Knowledge Advisor



Présentation
- Conventions

Nouveautés

Mise en route
- Utilisation des paramètres
- Utilisation des formules
- Utilisation des règles
- Suivant

Tâches de l'utilisateur
- Utilisation des paramètres
  - Création d'un paramètre
  - Présentation des paramètres
  - Copie et collage de paramètres
  - Spécification du paramètre Matériau
  - Spécification d'une valeur de paramètre en tant que mesure
  - Importation de paramètres
  - Création de points, de droites, etc. en tant que paramètres
  - Application de bornes à des paramètres
  - Création d'un lien associatif entre des mesures et des paramètres
  - Publication des paramètres
  - Familiarisation avec l'explorateur de paramètres
  - Ajout d'un paramètre à une entité
  - Ajout d'un paramètre à une arête
  - Verrouillage et déverrouillage d'un paramètre
  - Création d'ensembles de paramètres
  - Création d'un paramètre externe
  - Désactivation et réactivation de paramètres externes
- Utilisation de formules
  - Présentation des formules
  - Familiarisation avec la boîte de dialogue f(x)
  - Utilisation du dictionnaire
    - Constantes
    - Mots clés
    - Méthodes de table de paramétrage
    - Opérateurs
    - Points constructeurs
    - Messages et macros
    - Loi
    - Droites constructeurs

Cercles constructeurs
Fonction de recherche
Méthodes et fonctions relatives aux chaînes
Directions constructeurs
Fonctions utilisateur de l'atelier Electrical
Attribut
Liste
Mesures
Surfaces constructeurs
Wireframe constructeurs
Mesures de pièce
Plans constructeurs
Opérateurs d'analyse
Méthodes
Fonctions mathématiques
Space Analysis
Création d'une formule
Création de formules basées sur des publications
Spécification d'une mesure dans une formule
Utilisation d'une géométrie pour créer une formule
Référence à des paramètres externes dans une formule
Utilisation de la fonction Equivalence
Association d'adresses URL et de commentaires à des paramètres ou à des relations
Association d'adresses URL et de commentaires à des paramètres ou à des relations
Affectation d'un commentaire à une entité Knowledge
Utilisation des tables de paramétrage
Présentation des tables de paramétrage
Familiarisation avec la boîte de dialogue Table de paramétrage
Création d'une table de paramétrage à partir de valeurs des paramètres de document
Création d'une table de paramétrage à partir d'un fichier existant
Ajout d'une ligne au fichier externe d'une table de paramétrage en mode interactif
Contrôle de la synchronisation des tables de paramétrage
Stockage d'une table de paramétrage dans une copie optimisée
Création et utilisation d'une loi Knowledge Advisor
Utilisation de l'inspecteur de gestion de connaissance
Utilisation de la fonction Règle
Création d'une règle
Utilisation de règles et de vérifications dans une copie optimisée
Utilisation de l'éditeur de règles
Traitement des erreurs
Utilisation de la fonction Vérification
Création d'une vérification
Exécution d'une analyse globale des vérifications
Utilisation de l'outil d'analyse des vérifications
Présentation du rapport de vérification par défaut
Personnalisation des rapports de vérification
Utilisation de l'éditeur de vérifications
Utilisation de la fonction Réaction
Utilisation de la fenêtre Réaction

- Création d'une réaction : événement DragAndDrop
- Création d'une réaction : événement AttributeModification
- Création d'une réaction : événement Insert
- Création d'une réaction : événement Inserted
- Création d'une réaction : événement Remove
- Création d'une réaction : événement BeforeUpdate
- Création d'une réaction : événement ValueChange
- Utilisation d'une réaction avec un gabarit : événement Instantiation
- Utilisation d'une réaction Knowledge Advisor avec un modèle de document : événement Instantiation
- Création d'une réaction : événement Update
- Création d'une réaction : événement FileContentModification
- Création d'une boucle dans une réaction

Lancement d'une macro VB avec des arguments

Utilisation des relations

- Création d'ensembles de relations
- Utilisation des relations basées sur les publications au niveau du produit
- Activation et désactivation d'un composant
- Instanciation de relations issues d'un catalogue
- Mise à jour de relations utilisant des mesures
- Contrôle de la mise à jour des relations

Utilisation de la fonction Action

Utilisation de la fonction Liste

- Utilisation de la fenêtre Edition de liste
- Création d'une liste

Utilisation de la fonction Boucle

- Présentation de la fonction Boucle
- Familiarisation avec la fenêtre Edition de la boucle
- Déclaration des données d'entrée
- Définition du contexte
- Utilisation du langage de script
  - Structure du script d'actions
  - Propriétés d'objet
  - Mots clés
  - Variables
  - Opérateurs
  - Utilisation des commandes Insérer...
  - Commentaires
  - Limites
- Création d'une boucle
- Création d'une copie optimisée contenant une boucle

Résolution d'un ensemble d'équations

- Utilisation de l'éditeur d'équations

Utilisation du langage Knowledge Advisor

- Attributs

Limites

Conseils utiles

Exemples d'utilisation

- Roulement à billes

Préambule
Pas à pas
Système de trois équations dans trois variables

Interopérabilité de Knowledge Advisor

Utilisation optimale de CATIA PLM pour Knowledge Advisor
Sauvegarde d'une structure de produit contenant une règle dans ENOVIA LCA

Description de l'atelier

Barre de menus de Knowledge Advisor
Barre d'outils de Knowledge Advisor
Barre d'outils Objets réactifs
Barre d'outils Organisation de la connaissance
Barre d'outils Objets de contrôle
Barre d'outils Actions
Barre d'outils Outils
Barre d'outils Objets ensemble d'équations

Personnalisation de l'atelier Knowledge Advisor

Connaissance
Langage
Génération de rapport
Part Infrastructure for Knowledgeware Applications

Référence

Wireframe - Version de base
Objet GSMLine
Objet GSMCircle
Objet GSMPlane
Objet GSMPoint
Module Part Design
Module Part Shared
Objet congé de raccordement constant
cfyugpktScpConstantEdgeFillet.htm
Objet motif
Module standard
Module GSD Shared
Module GSD
Module Knowledge Expert
Module Mechanical Modeler

Glossaire

Index

Présentation

Ce document est destiné aux utilisateurs qui souhaitent se familiariser rapidement avec le produit Knowledge Advisor.

Cette présentation contient les informations suivantes :

- Présentation de Knowledge Advisor
- Avant de lire ce guide
- Pour tirer le meilleur parti de ce guide
- Accès aux documents exemples
- Conventions utilisées dans ce guide

Présentation de Knowledge Advisor

CATIA - KNOWLEDGE ADVISOR  est un produit *CATIA* qui permet aux utilisateurs d'intégrer des connaissances dans leurs conceptions et de les optimiser afin de faciliter les décisions techniques et de réduire les erreurs ou d'automatiser la conception pour une productivité maximale.

L'utilisateur peut intégrer des connaissances dans ses conceptions, par exemple des formules, des règles et des vérifications, puis les optimiser au moment opportun. Ces connaissances sont alors prises en compte et utilisées selon leur définition. Leur signification est également accessible. Par exemple, une tentative de vérification peut mettre en évidence les paramètres utilisés pour une vérification ; il est alors facile de comprendre rapidement de quelle manière une règle a été enfreinte.

En résumé, Knowledge Advisor permet à l'utilisateur :

- d'assimiler des connaissances en ingénierie, telles que les spécifications incorporées permettant une cohérence absolue ;
- de définir facilement et de partager des méthodes et des procédures avec les autres utilisateurs ;
- d'automatiser la définition de produit ;
- d'assurer le respect des règles de la société ;
- d'augmenter la productivité ;
- d'améliorer la gestion des connaissances à des fins de partage et de compréhension ;
- de mettre en place la gestion des composants de connaissance pour la personnalisation et la réutilisation ;
- de prendre en compte dès le début les spécifications relatives à la conception finale, afin d'éviter d'autres conceptions coûteuses ;

- d'être guidé et assisté pendant les tâches liées à la conception.

Avant de lire ce guide

Avant de lire ce guide, vous devez être familiarisé avec les concepts de base de CATIA version 5, tels que les fenêtres de document et les barres d'outils standard et d'affichage. Par conséquent, nous vous recommandons la lecture du document *Infrastructure - Guide de l'utilisateur* qui décrit les fonctionnalités génériques communes à tous les produits version 5. De plus, il décrit la présentation générale de CATIA V5 ainsi que l'interopérabilité entre les ateliers.

Pour tirer le meilleur parti de ce guide

Pour tirer le meilleur parti de ce guide, nous vous conseillons de lire le didacticiel Mise en route.

Lorsque vous aurez terminé, vous pourrez passer à la section suivante Tâches de l'utilisateur.

La section Description de l'atelier, qui décrit l'atelier Knowledge Advisor, et la section Personnalisation qui explique comment configurer les options, vont seront certainement utiles.

Accès aux documents exemples

Pour exécuter ces scénarios, vous utiliserez des documents exemples contenus dans le dossier online/kwrug/samples.

Pour plus d'informations, voir la section Accès aux documents exemples dans le document *Infrastructure - Guide de l'utilisateur*.

Conventions utilisées dans ce guide

Pour plus d'informations sur les conventions utilisées dans ce guide, voir la section *Conventions*.

Conventions utilisées

Les conventions utilisées dans la documentation CATIA, ENOVIA et DELMIA vous aideront à reconnaître un certain nombre de spécifications et de concepts importants.

Conventions graphiques

Les trois catégories de conventions graphiques suivantes sont utilisées :

- Conventions graphiques pour les éléments composant une tâche
- Conventions graphiques indiquant la configuration requise
- Conventions graphiques utilisées dans la table des matières

Conventions graphiques pour les éléments composant une tâche

Les conventions graphiques désignant les éléments composant une tâche ont les significations suivantes :

Cette icône...	Désigne...
	le temps nécessaire pour exécuter une tâche
	la cible affectée à une tâche
	les conditions préalables
	le démarrage du scénario
	un conseil
	un avertissement
	des informations
	des concepts de base
	une méthodologie
	des données de référence
	des informations relatives au paramètrage, à la personnalisation, etc.


la fin d'une tâche


les fonctionnalités nouvelles ou enrichies dans la présente édition


retour au mode d'affichage plein écran

Conventions graphiques indiquant la configuration requise

Les conventions graphiques désignant la configuration requise ont les significations suivantes :

Cette icône...	Indique les fonctions...
	propres à la configuration P1
	propres à la configuration P2
	propres à la configuration P3

Conventions graphiques utilisées dans la table des matières

Les conventions graphiques utilisées dans la table des matières ont les significations suivantes :

Cette icône...	Permet d'accéder à la rubrique...
	Plan du site
	Mode Affichage fractionné
	Nouveautés
	Généralités
	Mise en route
	Tâches de base
	Tâches de l'utilisateur ou Tâches avancées
	Interopérabilité
	Description de l'atelier
	Personnalisation
	Tâches d'administration



Informations de référence

Méthodologie

Foire aux questions (FAQ)

Glossaire

Index

Conventions de texte

Les conventions de texte suivantes sont utilisées :

- Les titres des documents CATIA, ENOVIA et DELMIA *apparaissent en italique* dans le texte.
- Les libellés des commandes apparaissent sous la forme **Fichier** -> **Nouveau**.
- Les fonctionnalités nouvelles ou enrichies sont signalées par un fond bleu.

Utilisation de la souris

L'utilisation de la souris diffère selon le type d'opération que vous devez effectuer.

Utilisez ce bouton...	Pour...
	• Sélectionner (un menu, une commande, une géométrie dans une zone graphique, etc.) • Cliquer (sur une icône, un bouton dans une boîte de dialogue, un onglet, un emplacement sélectionné dans la fenêtre du document, etc.) • Cliquer deux fois • Cliquer en maintenant la touche Maj enfoncée • Cliquer en maintenant la touche Ctrl enfoncée • Cocher (une case) • Faire glisser la souris • Faire glisser (une icône sur un objet, un objet sur un autre)
	• Faire glisser la souris • Déplacer



- Sélectionner un menu contextuel

Nouveautés

Nouvelle fonctionnalité

Affectation d'un commentaire à une entité Knowledge

Cette nouvelle fonctionnalité permet de modifier le texte affiché dans les info-bulles lorsque l'utilisateur fait glisser sa souris sur une entité Knowledge.

Mise en route

Avant de présenter les détails de l'utilisation de *CATIA* - Knowledge Advisor version 5, cette section fournit un scénario pas à pas expliquant comment utiliser les fonctions clé de Knowledge Advisor. Vous devez connaître les commandes de bases communes à tous les ateliers. Vous trouverez leur description dans le document *Infrastructure - Guide de l'utilisateur*.



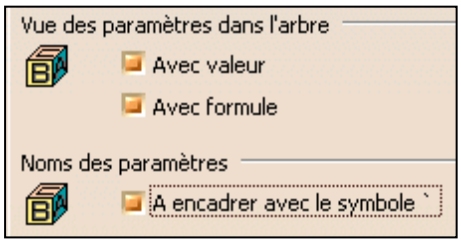
Lorsque vous travaillez dans un environnement japonais, pensez à sélectionner l'option **A encadrer avec le symbole '** (onglet **Outils->Options->Général->Paramètres et mesure->Knowledge**).

Utilisation des paramètres
Utilisation des formules
Utilisation des règles
Utilisation des vérifications

Utilisation des paramètres

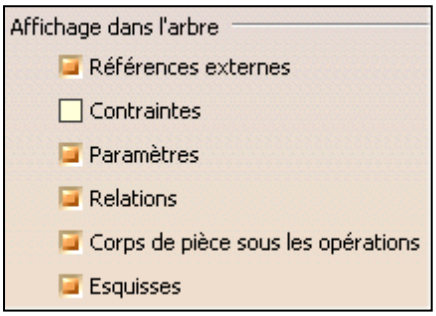
Dans cette tâche, vous apprendrez à utiliser des paramètres. Pour plus d'informations sur les tâches rattachées aux paramètres, reportez-vous au document Knowledgeware Infrastructure - Tips and Techniques - Summary consacré aux fonctions de règles de connaissance.

Cochez les paramètres ci-après :



- Dans le menu Outils, sélectionnez **Options->Général->Paramètres et mesure**.
- Dans l'onglet **Knowledge**, cochez les cases **Avec valeur** et **Avec formule**, puis cliquez sur **OK**.

Lorsque vous travaillez dans un environnement japonais, cochez la case **A encadrer avec le symbole '** située sous Noms des paramètres.

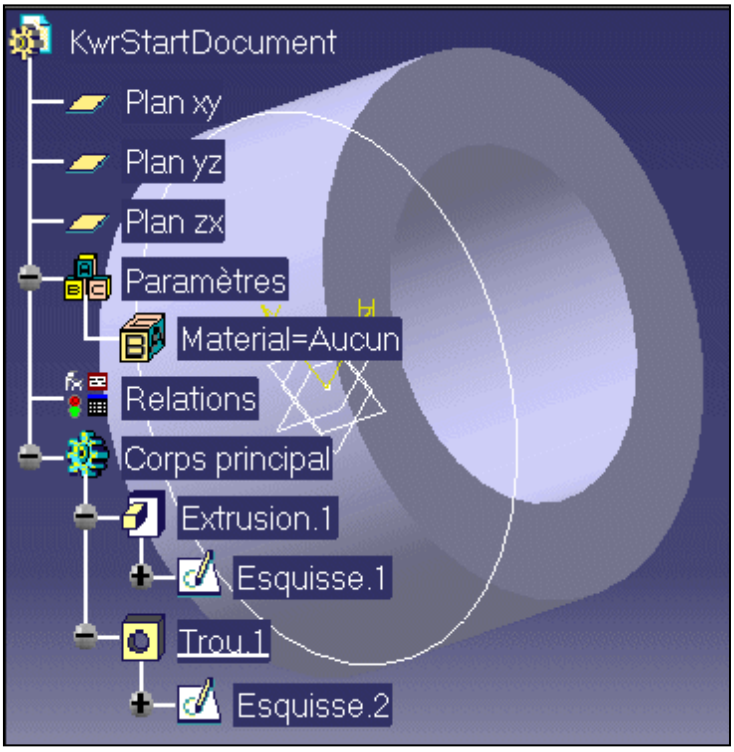


- Dans le menu **Outils**, sélectionnez **Outils->Options...->Infrastructure->Part Infrastructure.**
- Cochez au moins les cases **Relations** et **Paramètres** de l'onglet **Affichage**, puis cliquez sur **OK**.

Il est conseillé de cocher toutes les options situées sous les paramètres de l'arbre de spécifications.

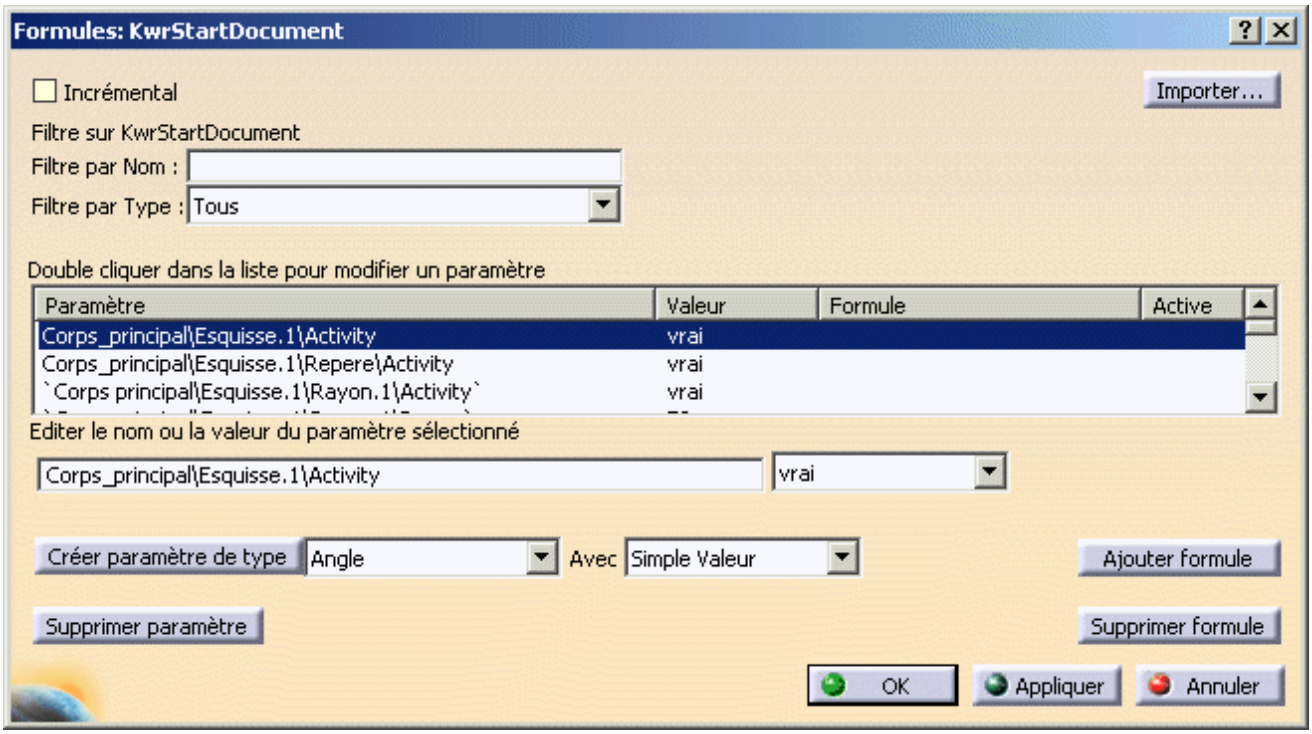
1. Ouvrez le document KwrStartDocument.CATPart.

Si vous développez le noeud Paramètres de l'arbre de spécifications, le paramètre Matériau est le seul à s'afficher.
A ce stade du scénario, ne prêtez pas attention à ce paramètre par défaut.



Le noeud Relations ne peut pas être développé dans la mesure où il n'existe pas de relation par défaut dans un document *CATIA*.

2. Cliquez sur l'icône f(x). La boîte de dialogue Formules s'affiche.



3. Dans la liste déroulante **Créer paramètre de type**, sélectionnez le type **Longueur**, puis cliquez sur le bouton **Créer paramètre de type**.

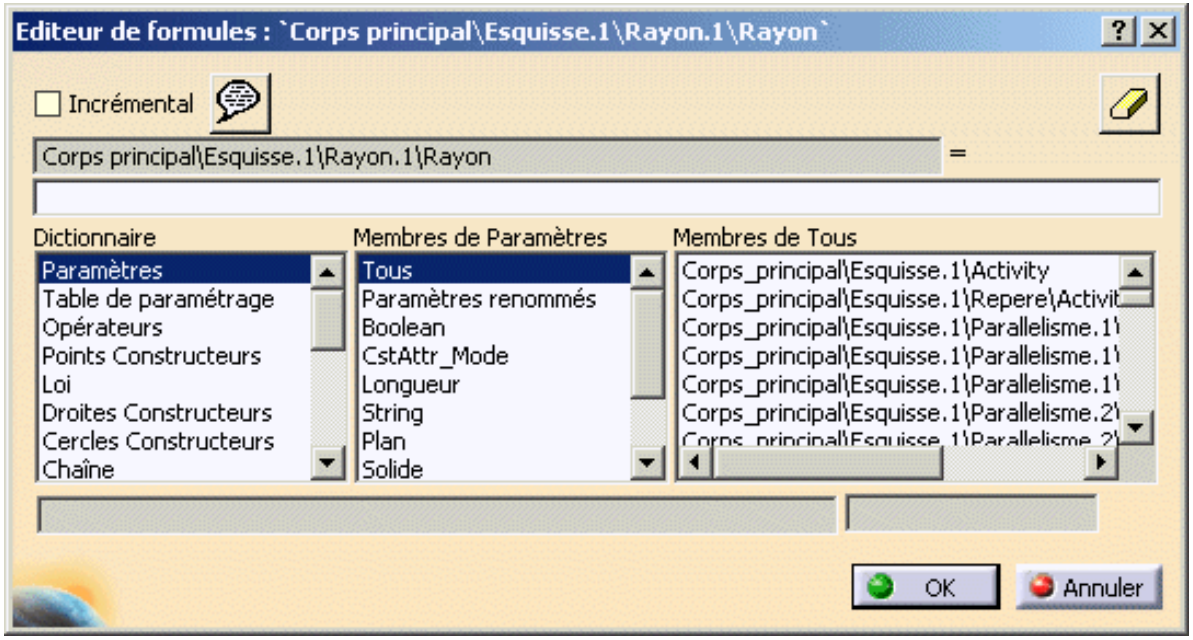
4. Dans le champ **Editer le nom ou la valeur du paramètre sélectionné**, remplacez la chaîne Length.1 par PadLength, puis cliquez sur **Appliquer**.
   Un nouveau paramètre apparaît à présent dans la liste des paramètres de la boîte de dialogue Formules et dans l'arbre de spécifications. Vous venez de créer un *paramètre utilisateur*.

5. Pour fermer la boîte de dialogue Formules, cliquez sur **OK**. Gardez le document ouvert et passez à la tâche suivante.

Utilisation des formules

Dans cette tâche, vous apprendrez comment associer une contrainte à un paramètre par le biais d'une formule.
Pour plus d'informations sur les formules, reportez-vous au document Knowledgeware Infrastructure - Tips and Techniques - Summary consacré à l'infrastructure des fonctions de règles de connaissance.

1. Cliquez sur l'icône  . La boîte de dialogue Formules s'affiche.
2. Dans la liste des paramètres, sélectionnez CorpsPrincipal\Esquisse.1\Rayon.1\Rayon, puis cliquez sur **Ajouter formule**. L'éditeur de formules s'affiche.



L'icône située à droite est simplement une gomme que vous pouvez utiliser pour effacer la formule.

3. Entrez la relation 2 * CorpsPrincipal\ Trou.1\ Diamètre.



4. Une fois que vous avez saisi la relation, cliquez sur **OK** dans l'**éditeur de formules**. La relation Formule.1 est ajoutée à l'arbre de spécifications.

   Dans la liste des paramètres de la boîte de dialogue, une formule est à présent associée au rayon de l'esquisse.

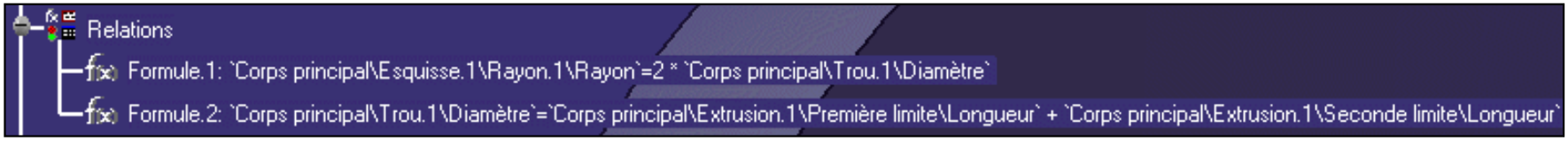
5. Dans la liste des paramètres, sélectionnez l'élément LongueurExtrusion, puis cliquez sur **Ajouter formule** pour créer la formule ci-dessous :

   ```
   LongueurExtrusion = CorpsPrincipal\Extrusion.1 \PremièreLimite \Longueur  + CorpsPrincipal\Extrusion.1 \SecondeLimite \Longueur
   ```

   Dans la liste des paramètres, la relation Formule.2 est à présent associée au paramètre utilisateur LongueurExtrusion. Dans l'arbre de spécifications, LongueurExtrusion apparaît également avec la valeur produite par Formule.2.

6. Cliquez deux fois sur **OK** dans la boîte de dialogue Formules pour terminer la tâche. Gardez le document ouvert et passez à la tâche suivante.

   Voici ce que vous devez voir maintenant dans l'arbre de spécifications, sous "Relations" ;

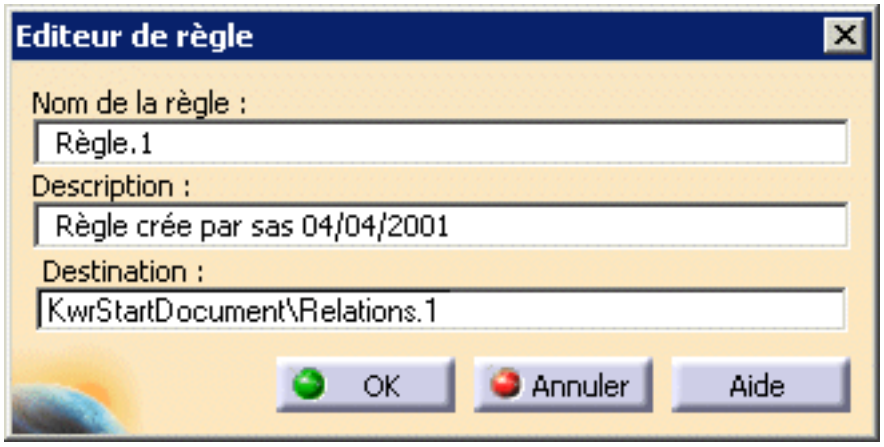


Utilisation des règles

Cette tâche présente les règles de Knowledge Advisor.
Contrairement aux fonctions liées aux paramètres et aux formules, qui sont accessibles à tous les utilisateurs de *CATIA*, les fonctions associées aux règles et aux vérifications requièrent le programme Knowledge Advisor.

Pour plus d'informations sur les règles, voir Utilisation de la fonction Règle.Pour plus d'informations sur l'éditeur de règles, voir Utilisation de l'éditeur de règles.

1. Sélectionnez l'élément KwrStartDocument dans l'arbre de spécifications.
2. Ouvrez l'atelier Knowledge Advisor à partir du menu **Démarrer->Knowledgeware**.
3. Cliquez sur l'icône de la règle  . La boîte de dialogue suivante s'affiche :



Les champs de la boîte de dialogue affichent les valeurs par défaut pouvant être modifiées :

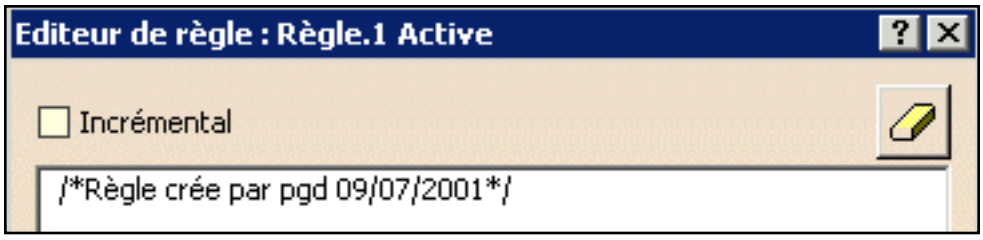
a - Le nom de la règle : Rule.1. Par défaut, la première règle créée dans un document est Rule.1.
Ce nom s'affiche dans l'arbre de spécifications à moins que vous ayez modifié le nom par défaut à sa création.

b - L'utilisateur et la date de création.

c - La cible, c'est-à-dire l'entité à laquelle vous allez ajouter la règle. Dans ce

scénario, la cible par défaut est l'entité Relations (le noeud Relations dans l'arbre de spécifications). Une règle peut toutefois être ajoutée à une autre entité et ne s'appliquer qu'à elle.

4. Remplacez la chaîne Règle.1 par RègleCylindre et, le cas échéant, modifiez les commentaires mais pas la cible. Cliquez sur **OK**. L'éditeur de règles s'affiche (comme illustré ci-dessous).



5. Tapez le code ci-dessous dans la zone d'édition des règles ou insérez-le à partir du navigateur avec un copier-coller.

```
CorpsPrincipal\Hole.1\Activité = vrai
if LongueurExtrusion <= 50mm and LongueurExtrusion >
20mm
{
PartBody\Hole.1\Diamètre = 20mm
Message("Le rayon externe est : # | Le rayon interne
est : #",
LongueurExtrusion,CorpsPrincipal\Hole.1\Diamètre)
}
else if LongueurExtrusion > 50mm and
LongueurExtrusion < 100mm
{
PartBody\Hole.1\Diamètre = 50mm
Message("Le rayon externe est : # | Le rayon interne
est : #",
LongueurExtrusion,CorpsPrincipal\Hole.1\Diamètre)
}
LongueurExtrusion >= 100mm
{
PartBody\Hole.1\Diamètre = 80mm
Message("Le rayon externe est : # | Le rayon interne
est : #",
LongueurExtrusion,CorpsPrincipal\Hole.1\Diamètre)
}
else
{
PartBody\Hole.1\Activité = faux
Message("Le rayon externe est : # | Le rayon interne
est : #",
LongueurExtrusion,CorpsPrincipal\Hole.1\Diamètre)
}
```

Les utilisateurs qui travaillent dans un environnement japonais doivent

utiliser le script suivant :

```
`CorpsPrincipal\Hole.1\Activité` = vrai
if `LongueurExtrusion` <= 50mm and
`LongueurExtrusion` > 20mm
{
`CorpsPrincipal\Hole.1\Diamètre` = 20mm
Message("Le rayon externe est : # | Le rayon interne
est : #",
`LongueurExtrusion`,`CorpsPrincipal\Hole.1\Diamètre`)
}
else if `LongueurExtrusion` > 50mm and
`LongueurExtrusion` < 100mm
{
`CorpsPrincipal\Hole.1\Diamètre` = 50mm
Message("Le rayon externe est : # | Le rayon interne
est : #",
`LongueurExtrusion`,`CorpsPrincipal\Hole.1\Diamètre`)
}
else if `LongueurExtrusion` >= 100mm
{
`CorpsPrincipal\Hole.1\Diamètre` = 80mm
Message("Le rayon externe est : # | Le rayon interne
est : #",
`LongueurExtrusion`,`CorpsPrincipal\Hole.1\Diamètre`)
}
else
{
`CorpsPrincipal\Hole.1\Activité` = faux
Message("Le rayon externe est : # | Le rayon interne
est : #",
`LongueurExtrusion`,`CorpsPrincipal\Hole.1\Diamètre`)
}
```

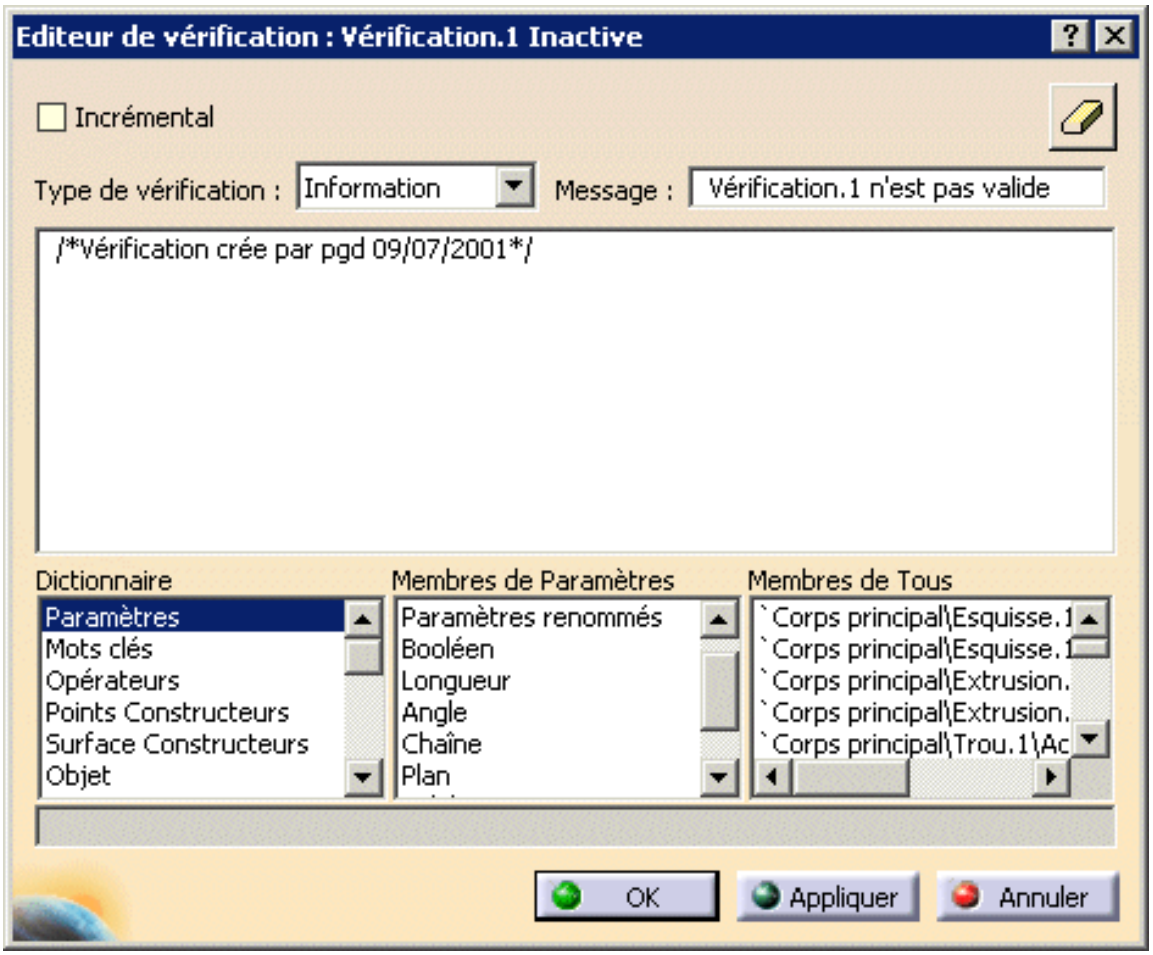
6. Cliquez sur **Appliquer**. Une fenêtre d'information affiche la valeur du paramètre utilisateur LongueurExtrusion et celle du diamètre interne de l'extrusion. Cliquez sur OK dans la fenêtre d'information. La relation RègleCylindre est ajoutée à l'arbre de spécifications.

7. Cliquez sur **OK** pour quitter cette partie de la boîte de dialogue. Gardez le document ouvert et passez à la tâche suivante.

Utilisation des vérifications

Dans cette tâche, vous apprendrez à créer une vérification et à comprendre ce qui se passe lorsque vous ajoutez une vérification au document. Le programme Knowledge Advisor est nécessaire pour cette tâche.
Pour plus d'informations sur les tâches rattachées aux vérifications, reportez-vous à la section Tâches de gestion des règles et des vérifications.

1. Cliquez sur l'icône . La première boîte de dialogue de l'éditeur de vérifications s'affiche.
2. Remplacez le nom par défaut Check.1 par VérifCylindre, puis cliquez sur **OK**. La boîte de dialogue **Editeur de vérification** s'affiche. Elle est semblable à celle de l'éditeur de règles. Vous devez désactiver la case **Incrémental**.



3. Sélectionnez Information dans la liste **Type de vérification**.
4. Entrez un texte dans la zone de message (par exemple : Extrusion trop courte). Ce message s'affichera chaque fois que la condition spécifiée par la vérification ne sera pas satisfaite.

5. Entrez l'instruction suivante dans la zone d'édition : LongueurExtrusion > 20mm.

6. Cliquez sur OK pour confirmer la création de la vérification. La relation VérifCylindre est ajoutée à l'arbre de spécifications. La présence d'une icône verte dans l'arbre de spécifications indique que la vérification a réussi. Aucun message ne s'affiche dans ce cas.

7. Entrez à présent l'instruction : LongueurExtrusion <= 20mm. La relation RègleCylindre est appliquée à nouveau. Une fenêtre d'information affiche la nouvelle valeur du paramètre utilisateur LongueurExtrusion et celle du diamètre interne de l'extrusion. Puis, un autre message vous avertit ("Extrusion trop courte") que la vérification n'est plus concluante. L'icône de vérification devient rouge dans l'arbre de spécifications.



Tâches de l'utilisateur

Pour connaître la liste complète des tâches liées aux règles et aux vérifications, voir Référentiel des tâches. Voir également les sections Conseils utiles, Limites et Infrastructure CATIA Knowledgeware.

Dans cette section, vous apprendre à gérer les relations Knowledge Advisor.

Utilisation des paramètres
Utilisation de formules
Association d'adresses URL et de commentaires à des paramètres ou à des relations
Utilisation des tables de paramétrage
Création et utilisation d'une loi Knowledge Advisor
Utilisation de l'inspecteur de gestion de connaissance
Utilisation de la fonction Règle
Utilisation de la fonction Vérification
Utilisation de la fonction Réaction
Lancement d'une macro VB avec des arguments
Utilisation des relations
Utilisation de la fonction Action
Utilisation de la fonction Liste
Utilisation de la fonction Boucle
Résolution d'un ensemble d'équations
Utilisation du langage Knowledge Advisor
Limites
Conseils utiles
Exemples d'utilisation

Utilisation des paramètres



Cliquez sur l'icône Formule pour créer des paramètres.



Cliquez sur l'icône Ajout d'ensemble de paramètres pour créer des ensembles de paramètres. Ces ensembles de paramètres sont tous regroupés sous le noeud Paramètres.



Cliquez sur l'icône Explorateur de paramètres pour ajouter de nouveaux paramètres à une entité.



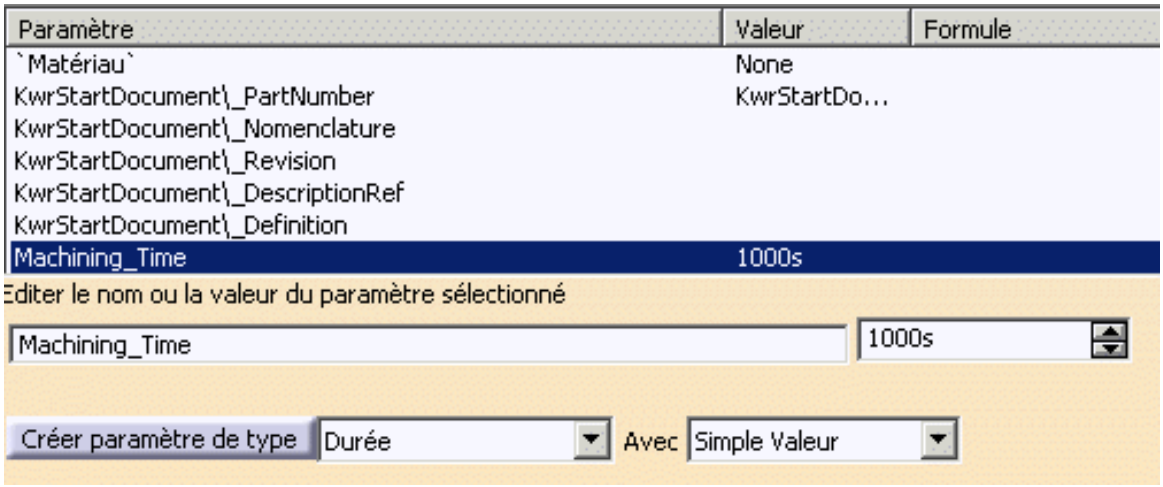
Cliquez sur l'icône Ajout de paramètres sur la géométrie pour ajouter de nouveaux paramètres à une face, à un sommet ou à une arête.

Création d'un paramètre
Présentation des paramètres
Copie et collage de paramètres
Spécification du paramètre Matériau
Spécification d'une valeur de paramètre en tant que mesure
Importation de paramètres
Création de points, de lignes... en tant que paramètres
Application de bornes à des paramètres
Création d'un lien associatif entre des mesures et des paramètres
Publication des paramètres
Familiarisation avec l'explorateur de paramètres
Ajout d'un paramètre à une entité
Ajout d'un paramètre à une arête
Verrouillage et déverrouillage d'un paramètre
Création d'ensembles de paramètres
Création d'un paramètre externe
Désactivation et réactivation de paramètres externes

Création d'un paramètre

Dans cette tâche, vous apprendrez à créer un paramètre de type Temps et à lui affecter une valeur.

1. Ouvrez le document KwrStartDocument.CATPart.
2. Cliquez sur l'icône f(x). La boîte de dialogue *f(x)* s'affiche.
3. Sélectionnez l'option **Durée** avec **Simple** dans la liste **Créer paramètre de type**, puis cliquez sur**Créer paramètre de type**. Le nouveau paramètre apparaît dans le champ Editer le nom ou la valeur du paramètre sélectionné.
4. Remplacez le nom Time.1 par Machining_Time et attribuez la valeur 1000 à ce paramètre. Cliquez ensuite sur **Appliquer**. Le paramètre Machining_Time est ajouté dans l'arbre de spécifications. La boîte de dialogue est modifiée ainsi :



5. Cliquez sur **OK** pour fermer la boîte de dialogue.

- Vous pouvez ajouter des propriétés à un document .CATPart ou .CATProduct en utilisant la commande Propriétés du menu contextuel. Il suffit de cliquer sur le bouton Autres propriétés dans l'onglet Produit, puis de cliquer sur Créer paramètre de type. La boîte de dialogue qui apparaît ressemble à la boîte de dialogue *f(x)*. Pour plus d'informations, reportez-vous au document *Product Structure - Guide de l'utilisateur*. Les propriétés que vous définissez ainsi sont également affichées dans la liste des paramètres de la boîte de dialogue *f(x)*.
- Vous pouvez indiquer qu'un paramètre est constant à l'aide de l'option Propriétés du menu contextuel. Cette option permet en outre de masquer un paramètre.

Présentation des paramètres

Lorsque vous créez une pièce telle que le cylindre creux de l'exemple "Mise en route", il vous arrive souvent de commencer par créer une esquisse, puis, à partir de cette dernière, de créer une extrusion. Pour finir, vous ajoutez d'autres entités à la nouvelle extrusion. Le document final se compose d'entités définissant ses propriétés intrinsèques. Si vous supprimez l'une de ces entités, le document sera modifié. Ces entités sont appelées ***paramètres***. Les paramètres jouent un rôle important dans les applications des règles de gestion de connaissance. Certaines entités peuvent être assorties de contraintes par le biais de relations, et peuvent également être utilisées en tant qu'arguments d'une relation.

Outre ces paramètres, *CATIA* vous permet de créer des ***paramètres utilisateurs***. Il s'agit d'informations supplémentaires qui sont ajoutées à un document.

Les paramètres utilisateurs sont très utiles dans les applications des règles de gestion de connaissance :

- Ils peuvent être utilisés pour ajouter à un document des informations spécifiques.
- Ils peuvent être définis ou assortis de contraintes par le biais de relations.
- Ils peuvent être utilisés en tant qu'arguments d'une relation.

Pour créer un paramètre, il vous suffit de cliquer sur l'une des icônes suivantes :



Les paramètres sont créés à l'aide de l'éditeur de formules. Pour plus d'informations sur cet éditeur, voir Familiarisation avec la boîte de dialogue ***f(x)***.



Les paramètres sont créés à l'aide de l'éditeur Explorateur de paramètres. Pour plus d'informations sur cet éditeur, voir Familiarisation avec l'explorateur de paramètres.



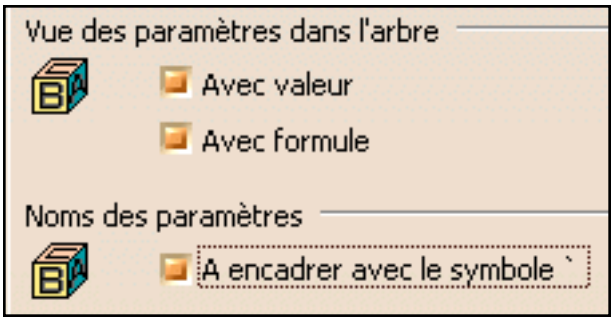
Les paramètres créés s'appliquent uniquement aux arêtes, aux faces et aux sommets. L'éditeur est semblable à l'éditeur Explorateur de paramètres.



L'icône Ensemble de paramètres permet de rassembler les paramètres utilisateur sous un ensemble.

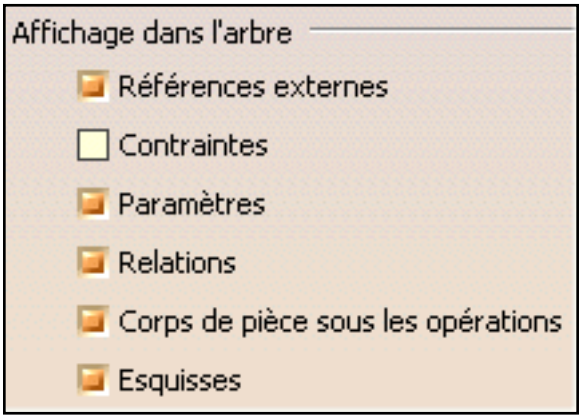
Une relation donnée peut prendre comme arguments ces deux types de paramètres (intrinsèques et utilisateurs).

Pour sélectionner les paramètres à afficher dans l'arbre de spécifications, cochez les options indiquées ci-dessous :



- Dans le menu Outils, sélectionnez Options->Général->Paramètres et mesures.
- Dans l'onglet Knowledge, cochez les cases Avec valeur et Avec formule, puis cliquez sur OK si vous souhaitez que les paramètres affichent les valeurs et les formules qui leur sont associées (le cas échéant).

Lorsque vous travaillez dans un environnement japonais, cochez la case A encadrer avec le symbole ' située sous Noms des paramètres.



- Dans le menu Outils, sélectionnez Outils->Options...->Infrastructure->Part Infrastructure.
- Cochez au moins les cases Relations et Paramètres de l'onglet Affichage, puis cliquez sur OK.

Copie/collage de paramètres

Avec les options **Outils -> Options -> Général -> Paramètres et mesure**, vous pouvez exécuter les actions suivantes :

- Coller un paramètre sans la formule qui le définit.
  Par exemple :
  Holeplus= 15 = Diamètre + 10 sera collée sous la forme `Réel.i = 15`
  (si l'option Avec valeur a été sélectionnée).

- Coller un paramètre ainsi que la formule qui le définit, mais uniquement si les paramètres auxquels la formule fait référence sont également sélectionnés dans la copie.
  Par exemple :
  Holeplus= 15 = Diamètre + 10 sera collée sous la forme `Réel.i = 15` si le paramètre Diamètre n'appartient pas aux éléments sélectionnés pour la copie,
  mais HolePlus sera collé sous la forme Réel.i = 15 = Réel.j + 10 si le paramètre Diamètre est sélectionné dans la copie (utilisez une sélection multiple).

- Coller un paramètre avec la formule.
  Holeplus= 15 = Diamètre + 10 sera collé sous la forme Réel.i = Diamètre + 10



Lors de la copie d'ensembles de paramètres contenant des paramètres cachés, ceux-ci sont collés automatiquement lors du collage des ensembles de paramètres et apparaissent comme des paramètres cachés.

Définition de la valeur du paramètre Matériau

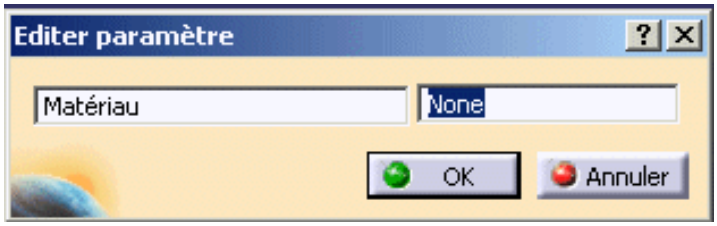
Quel que soit le document, le paramètre Matériau apparaît toujours dans l'arbre de spécifications. Il n'est créé qu'après l'application d'un matériau à une pièce ou un produit. Les propriétés mécaniques sont calculées sur la base de la valeur de ce paramètre. Définissez un matériau pour définir les valeurs des propriétés mécaniques.

1. Ouvrez le document KwrStartDocument.CATPart.

   Le paramètre Matériau s'affiche par défaut dans l'arbre de spécifications. Sa valeur est définie sur Aucun.



2. Cliquez deux fois sur le composant Matériau dans l'arbre de spécifications pour éditer le paramètre. La boîte de dialogue suivante s'affiche.

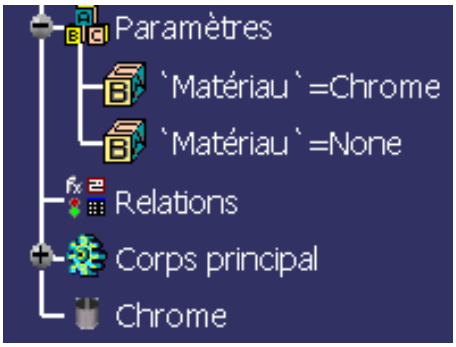


3. Cliquez sur **OK**, puis sélectionnez le composant principal dans l'arbre de spécifications.

4. Cliquez sur l'icône  dans la barre d'outils standard pour afficher la bibliothèque des matériaux disponibles. Sélectionnez le matériau **Matériau -> Fer**.

5. Cliquez sur **Appliquer des matériaux**, puis sur **OK**.

   Voici ce que vous devriez désormais voir dans l'arbre de spécifications. Le composant Fer (Iron) est ajouté dans l'arbre de spécifications et un nouveau matériau est ajouté sous le noeud Paramètres.



   N'oubliez pas que pour afficher les valeurs des paramètres, vous devez activer l'option **Outils -> Options -> Général -> Knowledge -> Paramètres et mesure -> Avec valeur**.

6. Gardez le document ouvert et passez à la tâche suivante.

Définition de la valeur des paramètres des propriétés mécaniques

Une fois définie la valeur du paramètre Matériau, les paramètres des propriétés mécaniques sont automatiquement mis à jour lorsque l'option Propriétés a été sélectionné dans le menu contextuel.

1. Sélectionnez l'élément principal dans l'arbre de spécifications et ouvrez la boîte de dialogue **Propriétés** à partir du menu contextuel.
2. Sélectionnez l'onglet **Masse**. Les propriétés mécaniques du document ont été mises à jour à partir de la valeur du paramètre Matériau.
3. Cliquez sur **OK** pour revenir au document.



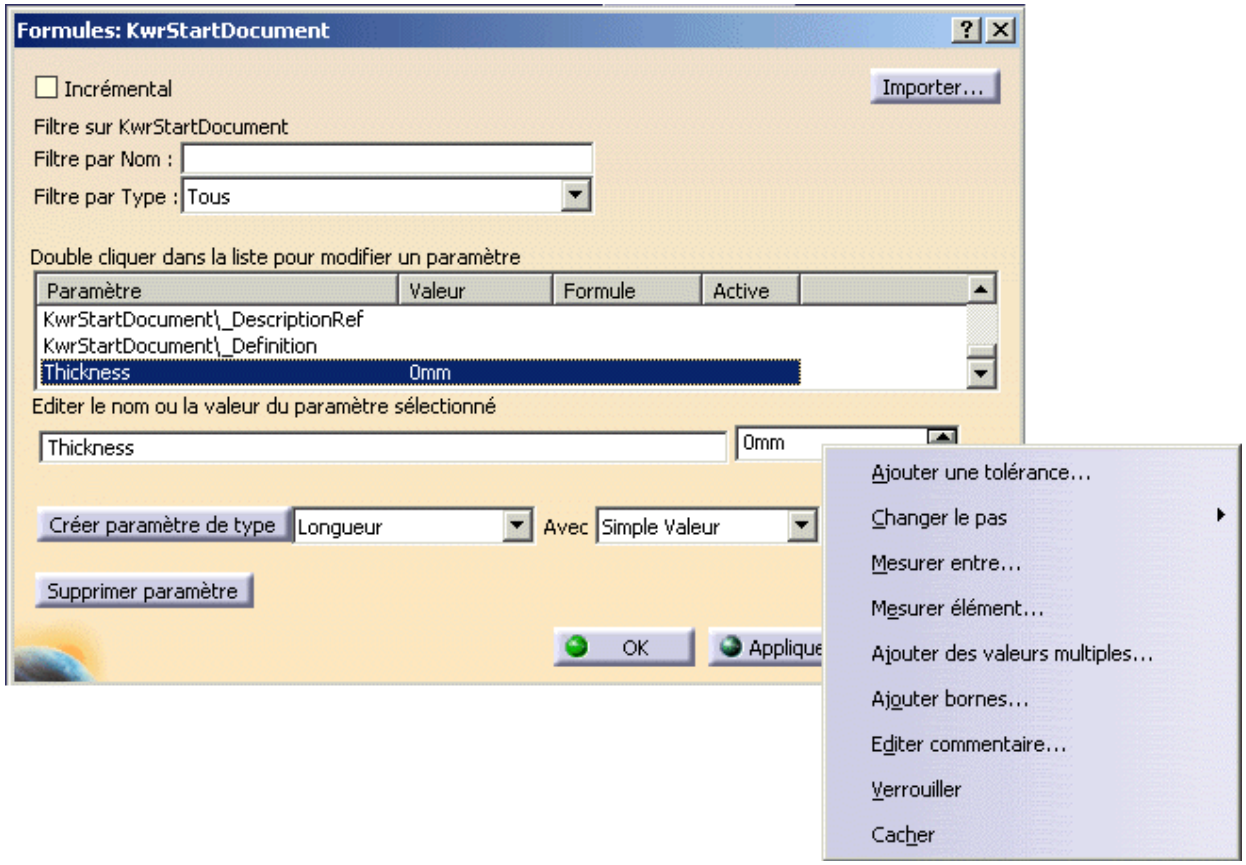
Définition d'une valeur de paramètre comme entité de mesure

Dans cette tâche, vous apprendrez à affecter une valeur à un paramètre, en la déduisant à partir d'une sélection graphique. Dans ce scénario, l'utilisateur déduit la valeur affectée au paramètre d'épaisseur en sélectionnant 2 arêtes circulaires.

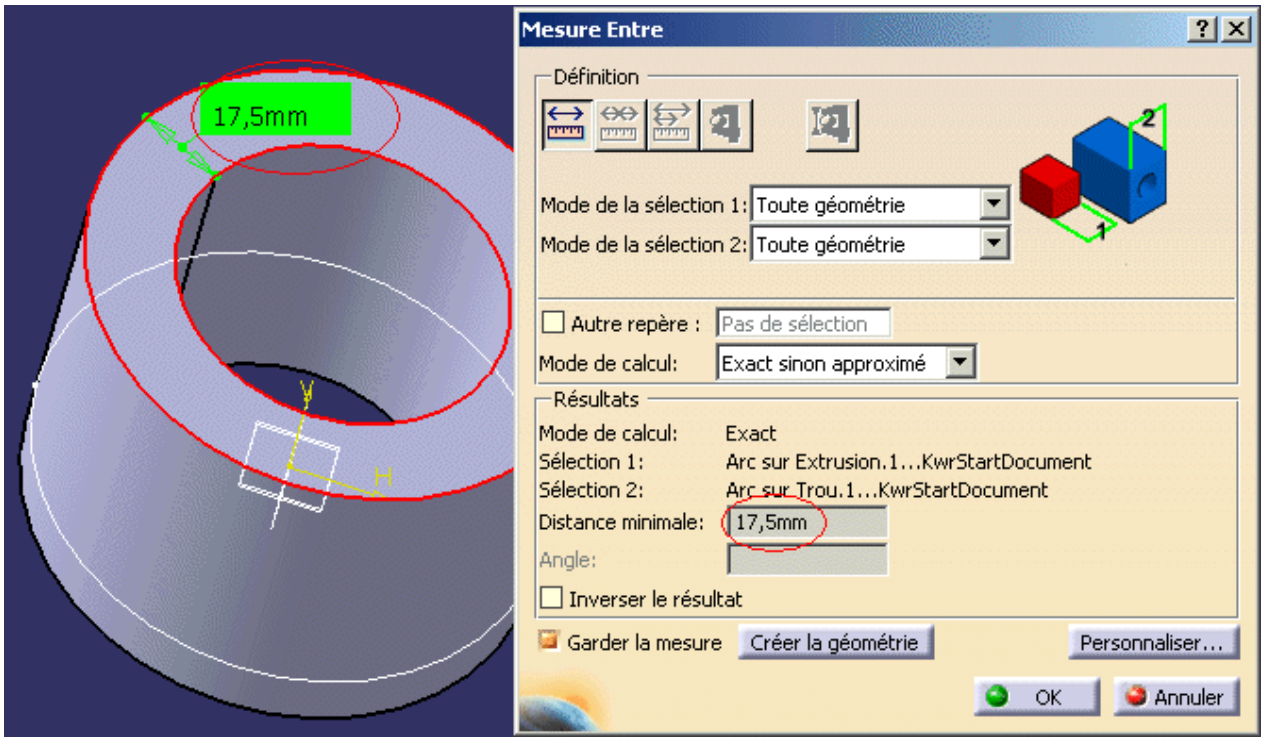
Pour affecter une valeur à un paramètre, il est courant d'utiliser le champ Editer le nom ou la valeur du paramètre sélectionné de la boîte de dialogue Formules. Il existe cependant une autre méthode. Il est possible de déduire un paramètre d'une sélection graphique.

1. Dans **Outils -> Options -> Général -> Paramètres et mesure**, activez la case **Chargement des bibliothèques d'extension du langage** de l'onglet Langage.
2. Ouvrez le document KwrStartDocument.CATPart.
3. Cliquez sur l'icône f(x). La boîte de dialogue *f(x)* s'affiche.
4. Sélectionnez l'option **Longueur** avec **Simple** dans la liste **Créer paramètre de type**, puis cliquez sur**Créer paramètre de type**.

   Le nouveau paramètre apparaît dans le champ Editer le nom ou la valeur du paramètre sélectionné.
5. Remplacez le nom Longueur.1 par Epaisseur, puis cliquez à l'aide du bouton droit de la souris dans le champ de valeur de **Editer le nom ou la valeur du paramètre sélectionné**.



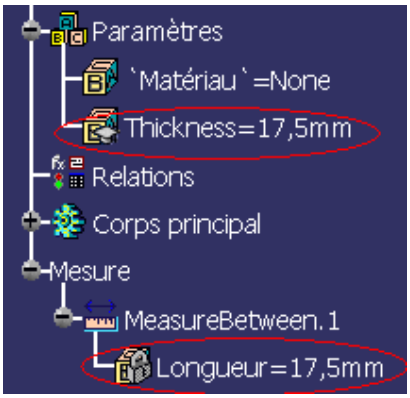
6. Sélectionnez la commande **Mesurer entre...** dans le menu contextuel. La boîte de dialogue Mesure entre s'affiche. Sélectionnez Ligne seule comme mode de **Sélection 1** et comme mode de **Sélection 2**.
7. Dans la zone géométrique du document, sélectionnez successivement l'une des arêtes circulaires internes de la pièce, puis l'arête circulaire externe sur la même face. La valeur 17.5 s'affiche dans la boîte de dialogue Mesure entre.



8. Dans la boîte de dialogue Mesure d'inertie, cliquez sur **OK**. La valeur 17.5 mm s'affiche dans la boîte de dialogue Formules.

9. Cliquez sur **OK** pour fermer la boîte de dialogue Formules.

Le paramètre s'affiche sous le noeud Mesure dans l'arbre de spécifications et sous le noeud Paramètres.



Pour éditer ce paramètre, procédez comme suit :

- Cliquez deux fois dessus dans l'arbre de spécifications. La boîte de dialogue Edition des paramètres s'affiche.
- Cliquez sur l'icône  située en regard du champ de valeur. La boîte de dialogue Mesure entre s'affiche.
- Editez le paramètre, puis cliquez sur **OK**.



Importation de paramètres

Dans cette tâche, vous apprendrez à importer des paramètres à partir d'un fichier Excel ou .txt dans un document CATPart.

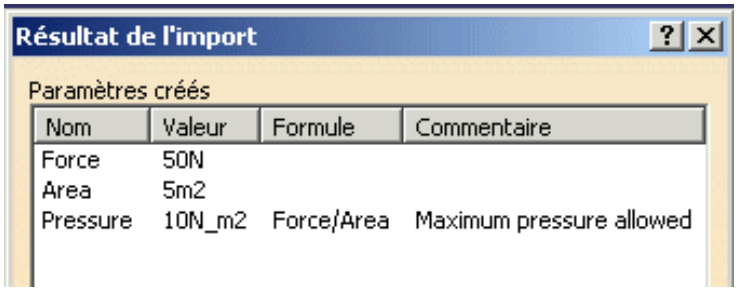
- Les paramètres et leurs valeurs peuvent être importés à partir d'un fichier texte ou d'un fichier Excel (Windows) dans des documents (CATPart, CATProduct, Drawings...).
- Si des paramètres importés existent déjà dans le document, l'importation actualise ce dernier automatiquement.
- Notez que si 2 paramètres portent le même nom dans le fichier d'importation, aucun clone n'est créé et les paramètres ne sont pas renommés. Avant de les importer, vérifiez que les paramètres portent des noms différents. Les noms de paramètre identiques ne sont pas pris en charge.

Vous trouverez ci-après les règles de formatage auxquelles doit se conformer le fichier externe :

- **Colonne 1**
  Nom des paramètres
- **Colonne 2**
  Valeurs de paramètre. *Plusieurs valeurs peuvent être définies*. Dans ce cas, elles doivent être séparées par un point virgule ";". La valeur importée est celle encadrée entre les signes "<" et ">". Utilisez la touche de tabulation pour passer d'une colonne à une autre dans un fichier texte avec tabulations.
- **Colonne 3**
  Formule. En l'absence de formule, la troisième colonne doit rester vide. Dans un fichier texte avec tabulations, pour laisser la colonne 3 vide, appuyez deux fois sur la touche de tabulation à partir de la colonne 2.
- **Colonne 4**
  Commentaire en option.

1. Ouvrez le document KwrStartDocument.CATPart.
2. Cliquez sur l'icône f(x). La boîte de dialogue *f(x)* s'affiche.
3. Cliquez sur Importer.... Une boîte de dialogue de sélection de fichier s'affiche.
4. Sélectionnez le fichier ExCompanyFile0.xls (Windows uniquement) ou le fichier TxCompanyFile0.txt, puis cliquez sur **Ouvrir**.

   La liste des paramètres à importer dans le document KwrStartDocument.CATPart s'affiche.



5. Cliquez sur **OK** pour importer les paramètres du fichier d'entrée dans le document KwrStartDocument.CATPart.

   Les paramètres importés s'affichent à présent dans la liste des paramètres de la boîte de dialogue *f(x)*

et dans l'arbre de spécifications.

Paramètre	Valeur	Formule
KwrStartDocument_DescriptionRef		
KwrStartDocument_Definition		
Force	50N	
Area	5m2	
Pressure	10N_m2	= Force/Area

6. Cliquez sur **OK** pour fermer la boîte de dialogue.

Création de points, de droites, etc. en tant que paramètres

Dans cette tâche, vous apprendrez à déterminer la position de l'axe d'inertie d'une extrusion. Pour ce faire, partez d'une extrusion, puis :

1. Créez une droite en utilisant la méthode ("Référence" ou )
2. Utilisez la droite constructeur inertiaAxis pour spécifier que cette droite doit être l'axe d'inertie de l'extrusion.
3. Extrayez les coordonnées du point situé à l'intersection de l'axe d'inertie et du plan de l'extrusion.

Pour créer des éléments du type Points, Droites, Courbes, Surfaces, Plans ou Cercles et les utiliser dans des relations Knowledgeware, vous pouvez :

- Créez ces éléments comme des éléments "isolés" dans l'atelier Generative Shape Design.Les éléments "isolés", également appelés *Référence*, sont des éléments dénués de liaisons avec les autres entités utilisées pour les créer. Pour plus d'informations sur les éléments de type "Référence", reportez-vous au document *Generative Shape Design - Guide de l'utilisateur*.

- Créez ces éléments à l'aide des fonctionnalités *f(x)* et sélectionnez le type d'élément correct dans la liste Créer paramètre de type.

1. Accédez à l'atelier Part Design, créez une esquisse dans le plan yz, puis extrudez-la. Si nécessaire, reportez-vous au document *Part Design - Guide de l'utilisateur*.
2. Créez une droite qui servira ultérieurement d'axe d'inertie.
3. Pour ce faire, cliquez sur l'icône Formules, sélectionnez l'élément Droite dans la liste Créer paramètre de type, puis cliquez sur Créer paramètre de type.
4. Cliquez sur l'icône Formules. Dans la liste des paramètres, sélectionnez la droite que vous venez de créer (Geometrical Set.1\Line.1).

5. Cliquez sur Ajouter formule et ajoutez la formule ci-dessous dans l'éditeur :

   ```
   Geometrical Set.1\Line.1 = inertiaAxis(3,PartBody)
   ```

   La fonction inertiaAxis est accessible à partir des Droites constructeurs. L'axe numéro 3 se trouve dans la direction de l'extrusion (normal à yz). Cliquez sur OK dans la boîte de dialogue Formules. L'axe d'inertie s'affiche dans la zone géométrique.

6. Revenez à  . Créez trois paramètres de type de longueur : X, Y et Z.

7. Extrayez les coordonnées du point situé à l'intersection de l'axe d'inertie et du plan "yz". Pour ce faire, créez les formules suivantes :

   ```
   X=intersect(Geometrical Set.1\Line.1, 'yz plane').coord(1)
   Y=intersect(Geometrical Set.1\Line.1, 'yz plane').coord(2)
   Z=intersect(Geometrical Set.1\Line.1, 'yz plane').coord(3)
   ```

   Vous accédez à la fonction "intersect" à partir des constructeurs Wireframe et à la méthode *point*.coord à partir de l'élément Mesures du dictionnaire.

8. Vérifiez la valeur affichée dans l'arbre de spécifications ainsi que dans la boîte de dialogue Formules.

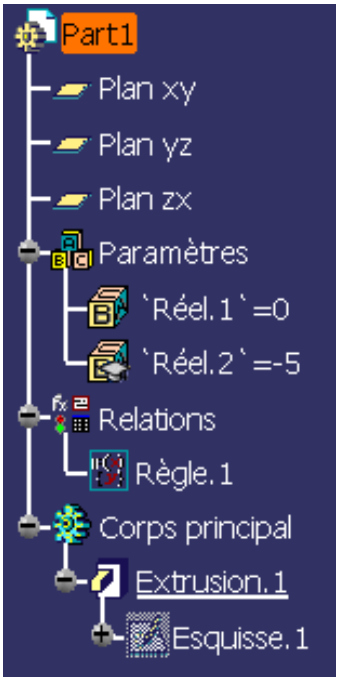
Le document KwoGettingStarted.CATPart, utilisé comme modèle dans le document *Engineering Optimizer - Guide de l'utilisateur*, illustre ce scénario.



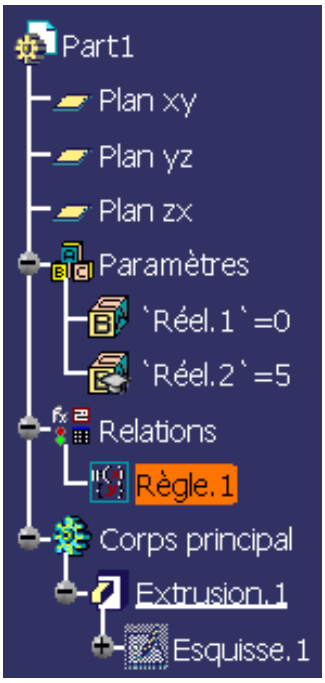
Application de bornes aux paramètres en utilisant une règle

Dans cette tâche, vous apprendrez à appliquer des bornes aux paramètres à l'aide d'une règle.

1. Ouvrez le document KwrRangesParameters.CATPart.
2. Cliquez sur l'icône  et sélectionnez **Réel** dans la liste déroulante pour créer deux paramètres de type "réel" : Réel.1 et Réel.2.
3. Sélectionnez Réel.1 et cliquez avec le bouton droit dans le champ se trouvant en regard de la zone **Editer le nom ou la valeur du paramètre sélectionné**.
4. Sélectionnez **Ajouter bornes ...** La boîte de dialogue **Bornes de Réel.1** apparaît.
5. Définissez les bornes minimum et maximum (-5 et 5, par exemple), puis cliquez deux fois sur **OK**.
6. Accédez à l'atelier Knowledge Advisor et cliquez sur l'icône Règle (). L'éditeur de règles s'affiche.
7. Entrez la règle suivante : **Réel.2 =Réel.1 .InferiorRange**, puis cliquez sur **OK**. La valeur de Réel.2 est remplacée par -5.



8. Cliquez deux fois sur la règle sous le noeud Relations et remplacez le script existant par **Réel.2 =Réel.1 .SuperiorRange**, puis cliquez sur**OK**. La valeur de Réel.2 est remplacée par 5.



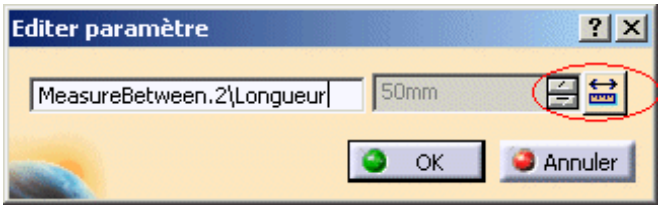
Création d'un lien associatif entre des mesures et des paramètres

Dans cette tâche, vous apprendrez à créer un lien persistant et associatif entre une mesure créée à l'aide de la commande **Mesurer élément** ou **Mesurer entre**, et un paramètre.

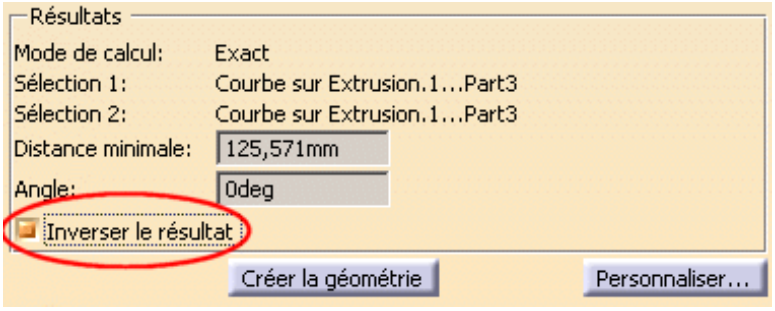
- **Mesurer élément** permet d'obtenir la longueur d'une courbe (arête, droite, courbe), le rayon ou l'angle d'un arc suivant la magnitude des paramètres.
- **Mesurer entre** permet d'obtenir la distance minimale ou l'angle entre deux éléments, suivant la magnitude des paramètres.

Il est possible de créer ce lien uniquement lorsque l'option **Garder la mesure** est activée dans les boîtes de dialogue **Mesure d'entités** et **Mesure entre** (sinon, le résultat est copié comme valeur simple).

- Aucune formule n'est créée lors de l'utilisation des commandes **Mesurer élément** ou **Mesurer entre**.
- L'icône située à droite du champ de l'éditeur est une icône de "mesure entre" ou de "mesure de l'élément". Notez que vous pouvez éditer la mesure.

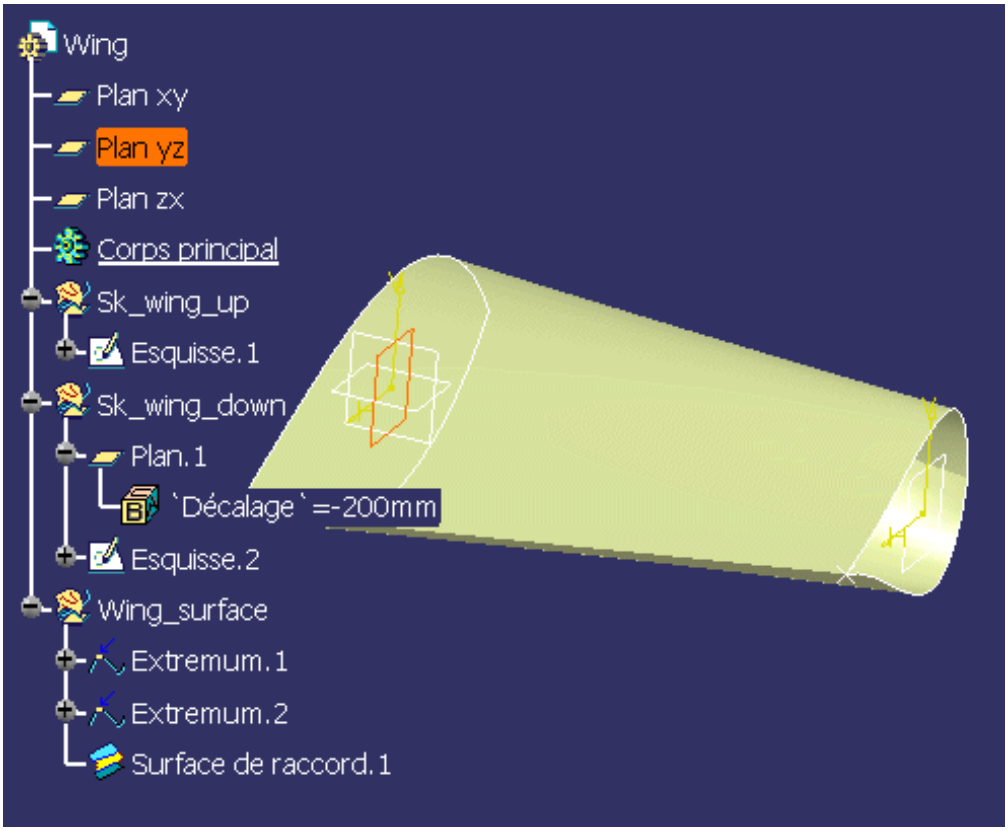


- Les paramètres situés sous le noeud Paramètres sont directement associés aux mesures.
- Vous pouvez inverser le signe du paramètre à l'aide de la commande **Inverser le résultat** dans la fenêtre Mesure d'entités ou Mesure entre. Le signe ne concerne que les paramètres définis et non celui de la mesure.



- Pour obtenir un lien associatif, vous devez effectuer une mesure associative. Si vous sélectionnez le mode **Point sur géométrie** et la fonction **Mesurer entre**, la mesure ne sera pas associative. En conséquence, il n'y aura aucune géométrie associative.
- Lorsqu'une mesure n'est pas associative, la valeur s'affiche dans le champ de valeur.
- Même en cas de mesure associative, si vous voulez uniquement obtenir le résultat de la mesure, désactivez l'option **Garder la mesure**.
- Pour créer une personnalisation "assistée", cliquez sur le bouton **Personnaliser...** dans la boîte de dialogue Mesure d'entités : vous pouvez alors visualiser les propriétés que le système peut détecter pour les divers types d'élément disponibles.

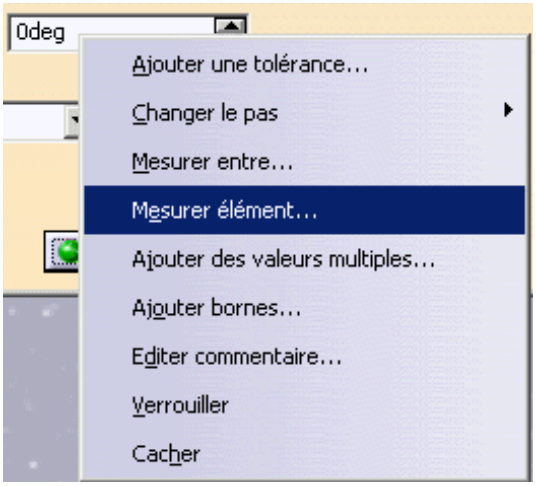
1. Ouvrez le fichier KwrPlaneWing.CATPart. L'image suivante s'affiche :



Utilisation de la commande Mesurer élément...

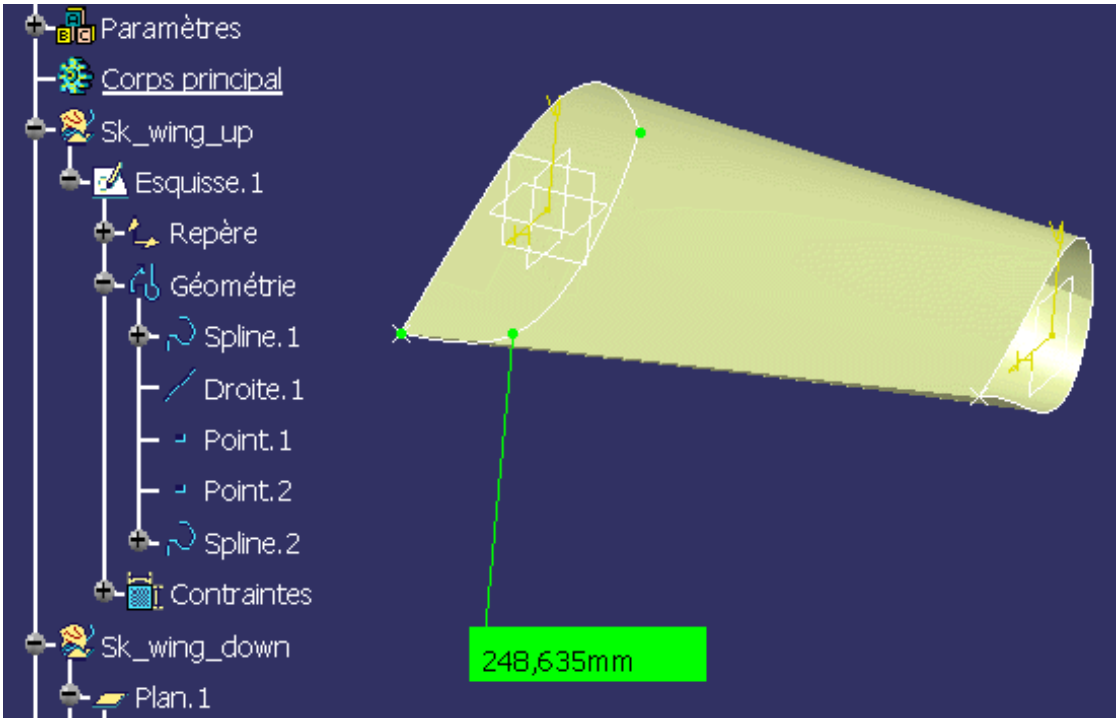
2. Ajoutez un paramètre de type **Longueur**. Pour ce faire, procédez comme suit :

   - Cliquez sur l'icône Formule( ). La boîte de dialogue Formules s'affiche.
   - Dans la liste déroulante **Créer paramètre de type**, sélectionnez **Longueur**, puis cliquez sur le bouton **Créer paramètre de type**.**Longueur.1** s'affiche dans le champ **Editer le nom ou la valeur du paramètre sélectionné**.
   - Cliquez à l'aide du bouton droit de la souris sur le champ de valeur de **Longueur.1** et sélectionnez la commande **Mesurer élément...**. La boîte de dialogue **Mesure d'entités** s'affiche.

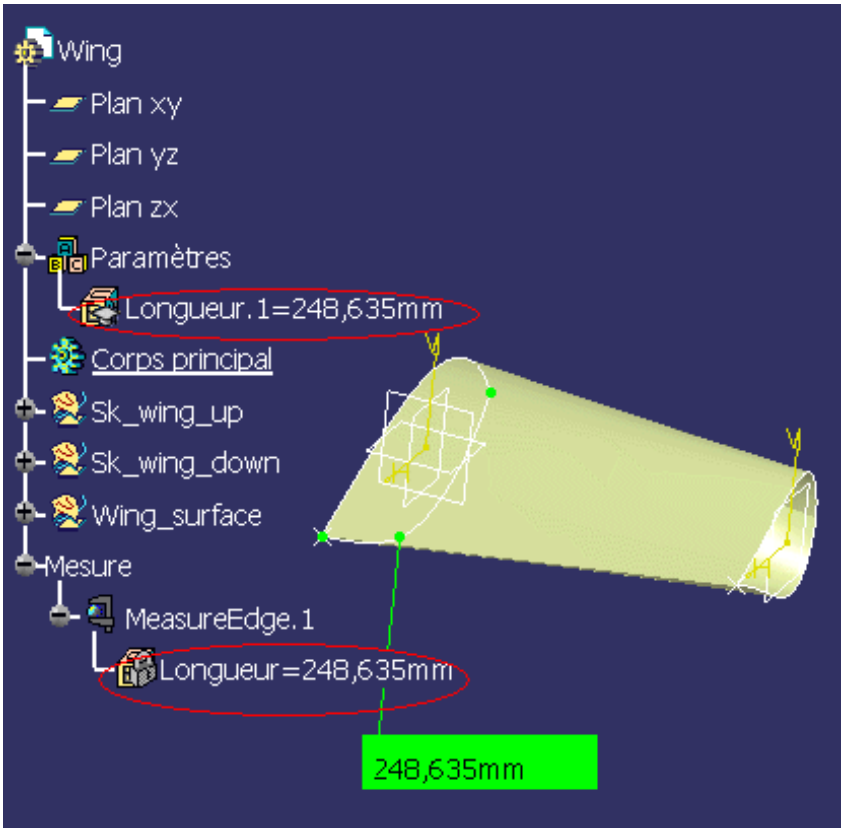


   - Vérifiez que l'option **Garder la mesure** est activée dans la boîte de dialogue **Mesure d'entités**.

- Dans l'arbre de spécifications, développez le noeud Esquisse.1 et sélectionnez Spline.2. L'élément sélectionné est mis en évidence dans la géométrie et sa mesure s'affiche en vert.



- Cliquez sur **OK** dans la boîte de dialogue **Mesure d'entités** ainsi que dans la boîte de dialogue Formules. Un nouveau paramètre est ajouté sous le noeud Paramètres et sous le noeud Mesure. Le paramètre Longueur.1 est désormais associé au résultat de la mesure.



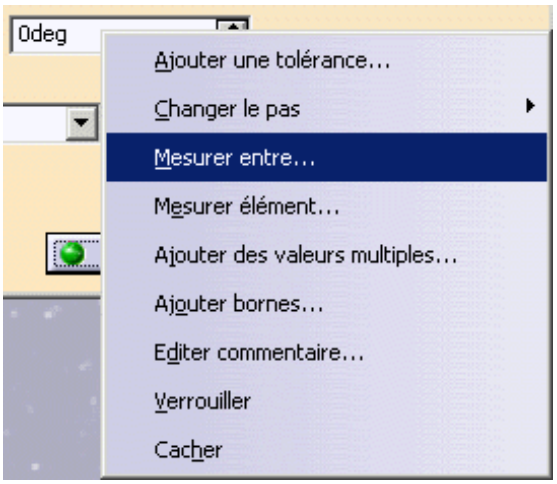
Utilisation de la commande Mesurer entre...

1. Ajoutez un paramètre de type **Angle**. Pour ce faire, procédez comme suit :

   - Cliquez sur l'icône Formule( ). La boîte de dialogue Formules s'affiche.

   - Dans la liste déroulante **Créer paramètre de type**, sélectionnez **Angle**, puis cliquez sur le bouton **Créer paramètre de type**. **Angle.1** s'affiche dans le champ **Editer le nom ou la valeur du paramètre sélectionné**.

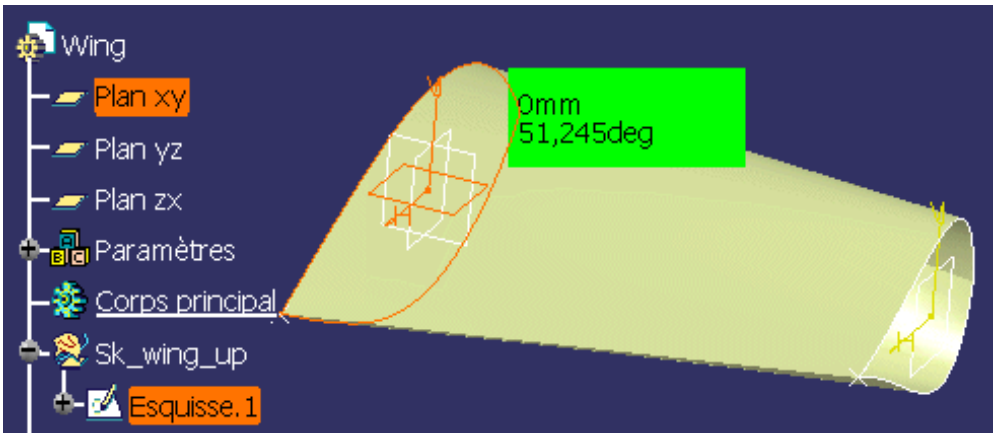
   - Cliquez à l'aide du bouton droit de la souris sur le champ de valeur de **Angle.1** et sélectionnez la commande **Mesurer entre...**. La boîte de dialogue **Mesure entre** s'affiche.



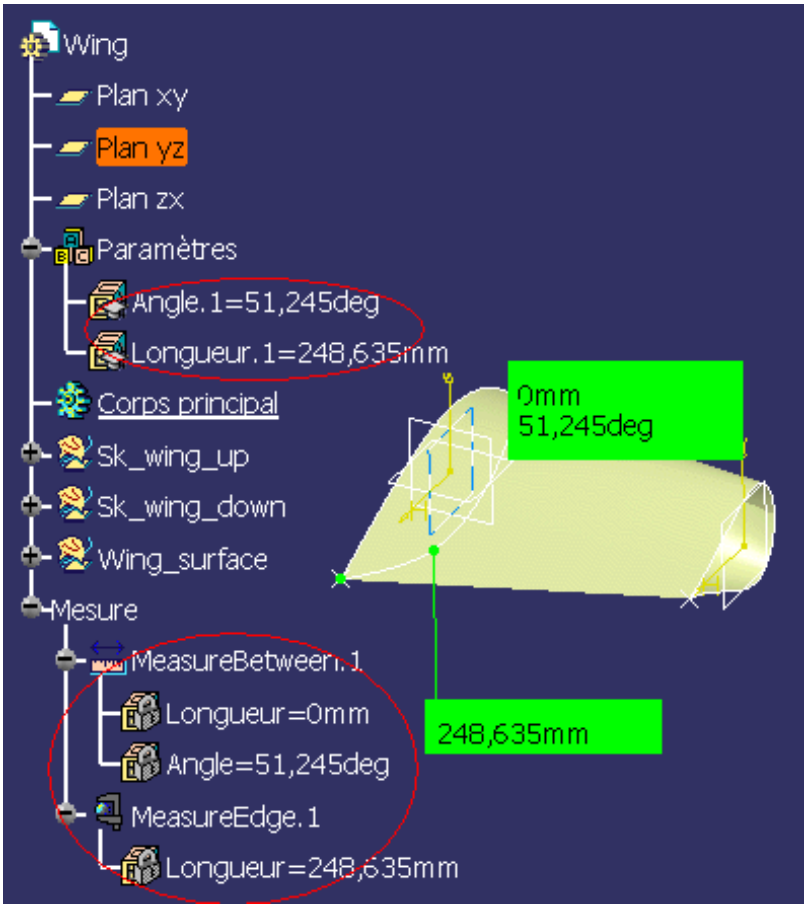
   - Dans la liste déroulante **Mode de la sélection 2**, sélectionnez l'option **Ligne seule**.

   - Dans l'arbre de spécifications, sélectionnez Plane xy, puis la géométrie, comme indiqué ci-après.

   Les éléments sélectionnés sont mis en évidence dans la géométrie et la mesure s'affiche en vert.



   - Cliquez sur **OK** dans la boîte de dialogue **Mesure entre** ainsi que dans la boîte de dialogue Formules. Un paramètre d'angle est ajouté sous le noeud Paramètres tandis que la mesure s'affiche sous le noeud Mesure.



Cliquez ici pour afficher le modèle obtenu.

Remarquez les situations suivantes :

- si plusieurs caractéristiques de la mesure sont calculées et présentent la même magnitude, le système choisit la plus pratique selon les règles prédéfinies.
- Pour supprimer le lien à la mesure, cliquez avec le bouton droit de la souris sur l'élément de mesure dans l'arbre de spécifications et sélectionnez la commande **objet mesure -> Supprime le lien avec la mesure**.

Publication de paramètres

Dans cette tâche, vous apprendrez à publier des paramètres. Le scénario ci-après se compose des étapes suivantes :

- Ajout des paramètres au document Screw.2 et publication des paramètres de diamètre, profondeur et volume. Répétition de ces opérations avec le deuxième fichier CATPart.
- Création d'un fichier CATProduct et importation de Screw.2.
- Dans le contexte du boulon, insertion de l'écrou qui importe les paramètres de profondeur et de diamètre en choisissant les publications MyDepth et MyDiameter de Screw.2.
- Dans le contexte du boulon, remplacement de Screw.2 (KwrScrew.CATPart) par Screw.2 (KwrScrew2.CATPart) dont la structure est différente du premier, mais qui possède les mêmes publications. Les deux paramètres et la vérification sont recalculés.

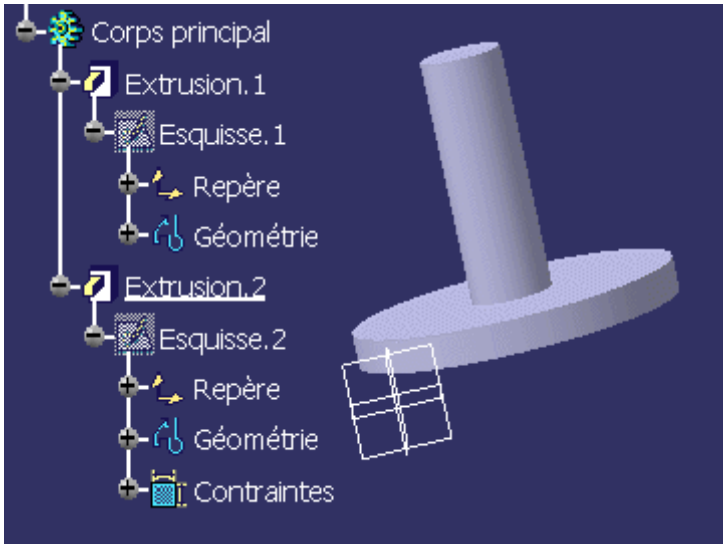
Une publication dispose d'un nom et fait référence à une géométrie ou à des paramètres d'un produit (ou de l'un de ses sous-produits).

La publication de paramètres doit être utilisée dans les situations suivantes :

- Définition d'une importation de paramètres entre deux pièces (similaire à l'importation de géométrie).
- Définition de relations au niveau de l'assemblage entre des paramètres (similaire aux contraintes).

Avant d'exécuter ce scénario, assurez-vous que l'option **Garder le lien avec l'objet sélectionné** est activée (**Outils -> Options -> Infrastructure -> Part Infrastructure -> Général**).

1. Ouvrez le document KwrScrew.CATPart. L'image suivante s'affiche.



2. Ajoutez des paramètres à la pièce. Pour ce faire, procédez comme suit :

- Cliquez sur l'icône  . L'éditeur de formules s'ouvre.Dans la liste déroulante **Créer paramètre de type**, sélectionnez **Volume**, puis cliquez sur le bouton **Créer paramètre de type**.

- Dans le champ **Editer le nom ou la valeur du paramètre sélectionné**, entrez le nom du paramètre : **MyVolume**. Cliquez sur **Appliquer**, puis sur le bouton **Ajouter formule**. L'éditeur de formules s'ouvre.

- Entrez la formule suivante à l'aide du dictionnaire : **smartVolume(PartBody\Pad.1) + smartVolume(PartBody\Pad.2)** . Cliquez sur **OK**, puissur **Oui**.

- Dans la liste déroulante **Créer paramètre de type**, sélectionnez **Longueur**, puis cliquez sur le bouton **Créer paramètre de type**.

- Dans le champ **Editer le nom ou la valeur du paramètre sélectionné**, entrez le nom du paramètre : **MyDepth**. Cliquez sur **Appliquer**, puis sur le bouton **Ajouter formule**.

- Entrez la formule suivante : **MyDepth=PartBody\Pad.2\FirstLimit\Length** et cliquez sur **OK**.

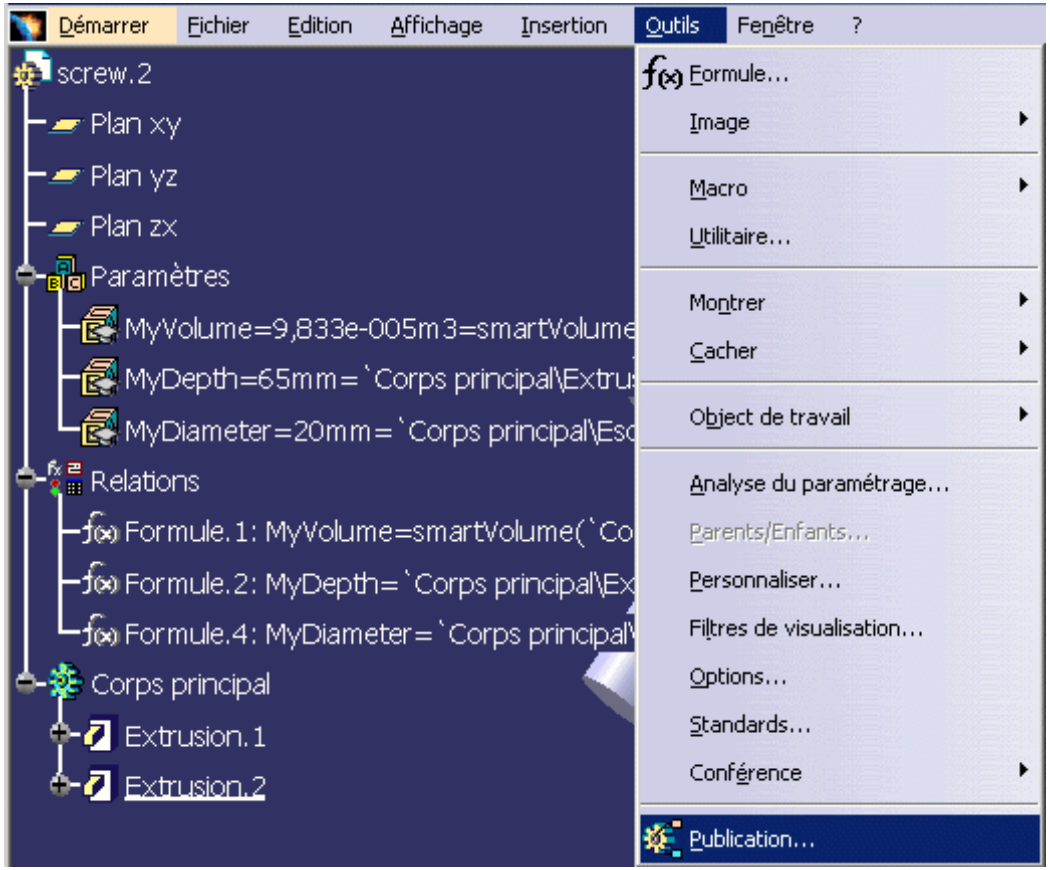
- Dans la liste déroulante **Créer paramètre de type**, sélectionnez **Longueur**, puis cliquez sur le bouton **Créer paramètre de type**.

- Dans le champ **Editer le nom ou la valeur du paramètre sélectionné**, entrez le nom du paramètre : **MyDiameter**. Cliquez sur **Appliquer**, puis sur le bouton **Ajouter formule**.

- Entrez la formule suivante : **MyDiameter=PartBody\Esquisse.2\Rayon.2\Rayon * 2**. Cliquez sur **OK** deux fois.

3. Publiez les paramètres **MyVolume**, **MyDepth** et **MyDiameter**.

Pour ce faire, sélectionnez la commande**Outils -> Publication** et cliquez sur les paramètres **MyVolume**, **MyDepth** et **MyDiameter** dans l'arbre de spécifications. Cliquez sur **OK**. Les paramètres publiés apparaissent dans l'arbre de spécifications sous le noeud Publication. Fermez le fichier.

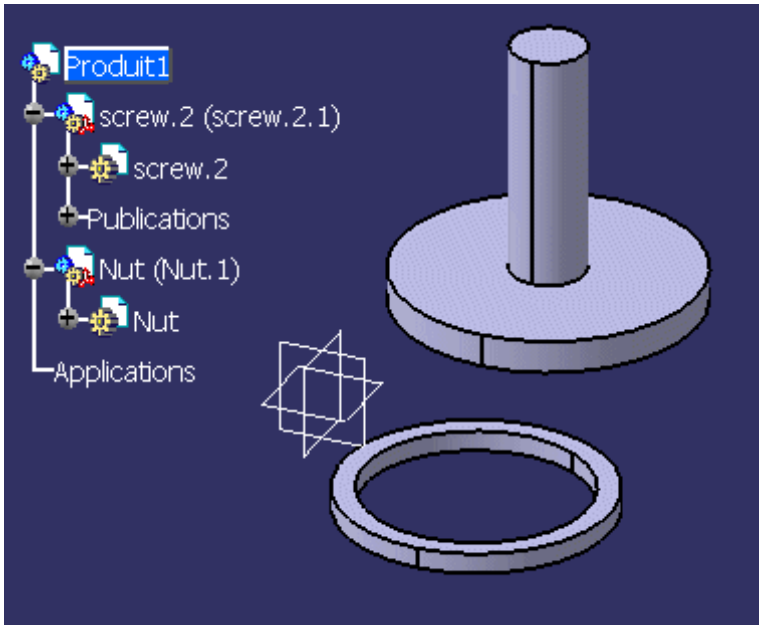


4. Ouvrez le document KwrScrew2.CATPart et répétez les étapes ci-dessus (1 à 3 incluses). La pièce doit être identique à celle ci-dessous. Fermez le fichier.



5. Créez un fichier CATProduct. Choisissez la commande **Insertion -> Composant existant...** et cliquez sur la racine de l'arbre de spécifications. La boîte de dialogue **Sélection de fichier** s'affiche. Sélectionnez le fichier KwrScrew.CATPart et cliquez sur **Ouvrir**. La vis est importée.

6. Choisissez la commande **Insertion -> Composant existant...**, sélectionnez le fichier Kwrnut.CATPart et cliquez sur **Ouvrir**. L'écrou est inséré.



7. Cliquez deux fois sur le cercle intérieur de l'écrou : la fenêtre Définition du trou s'affiche.

   - Cliquez à l'aide du bouton droit de la souris dans le champ **Diamètre** et choisissez la commande **Editer formule...** . L'éditeur de formules s'ouvre.

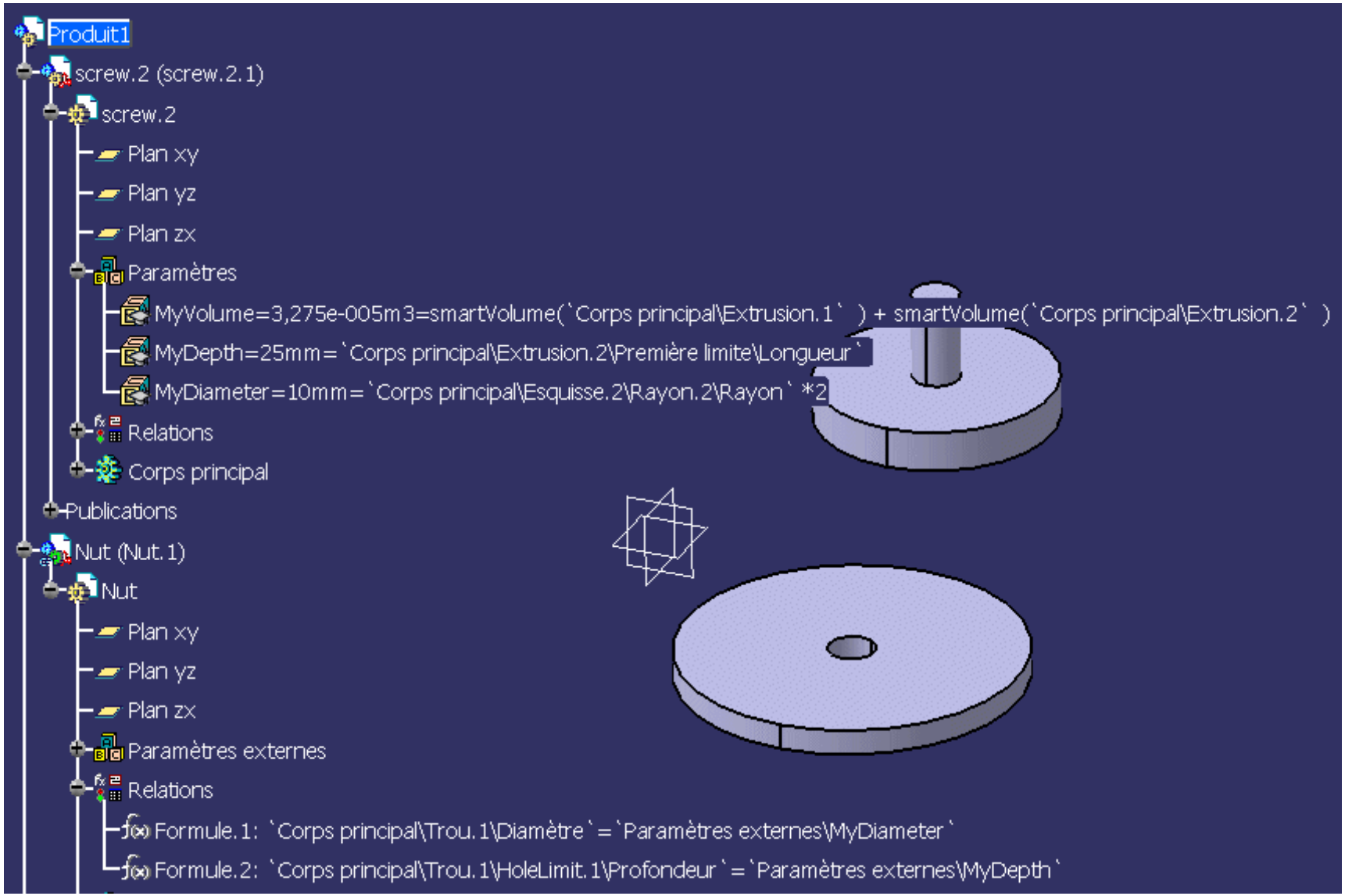
   - Sélectionnez MyDiameter dans les publications de la vis (screw). La formule doit être la suivante : **PartBody\Hole.1\Diameter=`Paramètres externes\MyDiameter`**. Cliquez sur **OK**.

   - Cliquez à l'aide du bouton droit de la souris dans le champ **Profondeur** et choisissez la commande **Editer formule...** . L'éditeur de formules s'ouvre.

   - Sélectionnez MyDepth dans les publications de la vis (screw). La formule doit être la suivante : **PartBody\Hole.1\HoleLimit.1\Depth=`Paramètres externes\MyDepth`**. Cliquez deux fois sur **OK**.

8. Cliquez deux fois, puis cliquez avec le bouton droit de la souris sur le composant Screw.2 dans l'arbre de spécifications et choisissez l'option **Composants -> Remplacer le composant...**. La boîte de dialogue **Sélection de fichier** s'affiche. Sélectionnez le fichier KwrScrew2.CATPart et cliquez sur **Ouvrir**.

9. Cliquez sur **Oui** lorsque vous êtes invité à préciser si vous voulez remplacer toutes les instances avec la même référence que le produit sélectionné. Mettez à jour l'écrou : les paramètres sont recalculés.


Produit1
screw.2 (screw.2.1)
screw.2
Plan xy
Plan yz
Plan zx
Paramètres
MyVolume=3,275e-005m3=smartVolume(`Corps principal\Extrusion.1`) + smartVolume(`Corps principal\Extrusion.2`)
MyDepth=25mm=`Corps principal\Extrusion.2\Première limite\Longueur`
MyDiameter=10mm=`Corps principal\Esquisse.2\Rayon.2\Rayon` *2
Relations
Corps principal
Publications
Nut (Nut.1)
Nut
Plan xy
Plan yz
Plan zx
Paramètres externes
Relations
Formule.1: `Corps principal\Trou.1\Diamètre`=`Paramètres externes\MyDiameter`
Formule.2: `Corps principal\Trou.1\HoleLimit.1\Profondeur`=`Paramètres externes\MyDepth`

Familiarisation avec l'explorateur de paramètres

Contrairement aux paramètres créés à l'aide de l'éditeur de formules, les paramètres créés à l'aide de l'explorateur de paramètres s'affichent sous l'entité sélectionnée dans l'arbre de spécifications.

La boîte de dialogue Explorateur de paramètres s'affiche lorsque vous cliquez sur l'icône  dans la barre d'outils standard. Cette boîte de dialogue vous permet d'ajouter des paramètres aux entités. Elle est composée des champs suivants :

- Entité
- Paramètres
- Paramètre
- Propriétés
- Bornes

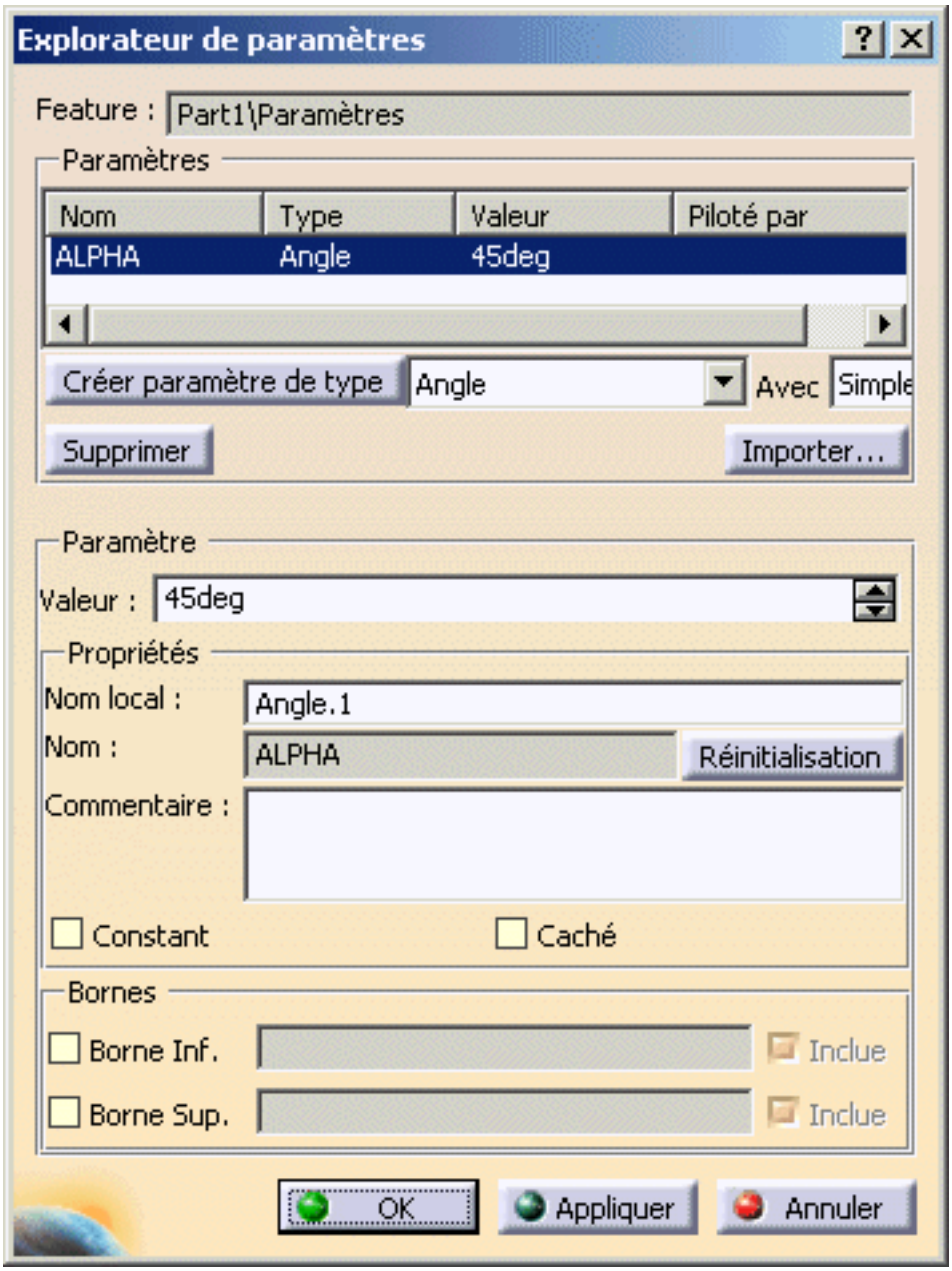
Entité

Ce champ indique l'élément sélectionné dans l'arbre de spécifications auquel le paramètre sera ajouté.

Paramètres

Ce champ permet de créer les paramètres qui seront ajoutés à l'entité sélectionnée dans l'arbre de spécifications. Pour plus d'informations sur ce champ, reportez-vous à la section Familiarisation avec la boîte de dialogue ***f(x)***.

Paramètre



Le champ **Valeur** permet d'affecter une valeur au paramètre créé.

Propriétés

Le champ **Nom local** permet de modifier le nom du paramètre créé.

Le champ **Nom** indique la manière dont le paramètre s'affichera dans les éditeurs.

Le champ **Commentaire** permet d'ajouter des commentaires au paramètre.

Une fois sélectionnée, la case **Constant** permet de verrouiller le paramètre. Il est alors impossible de le modifier.

Une fois sélectionnée, la case **Caché** permet de choisir d'afficher ou non le paramètre.

Bornes

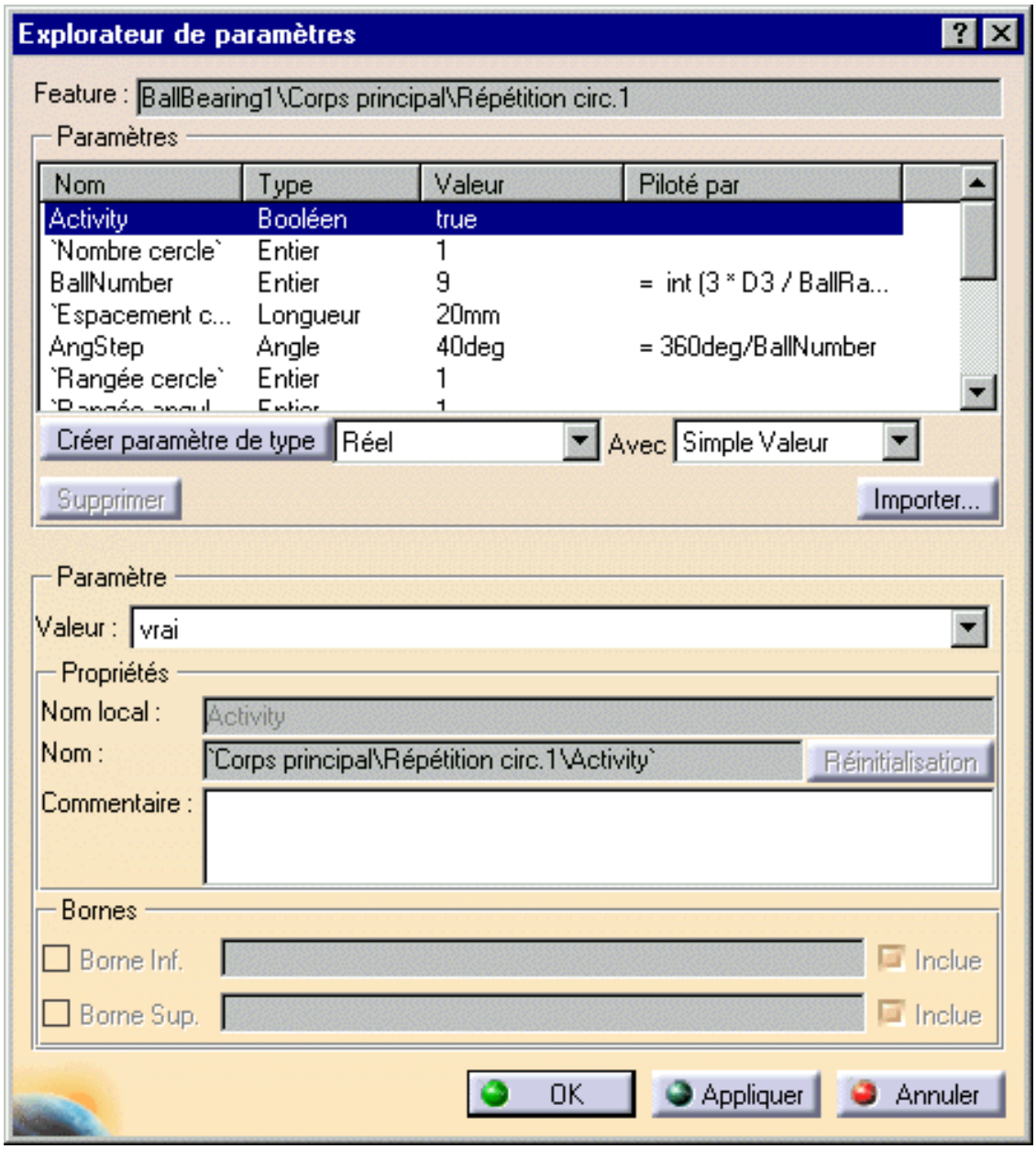
Une fois sélectionnée, la case **Borne Inf.** permet d'ajouter une borne inférieure au paramètre.

Une fois sélectionnée, la case **Borne Sup.** permet d'ajouter une borne supérieure au paramètre.

Ajout d'un paramètre à une entité

Dans cette tâche, vous allez apprendre à ajouter deux paramètres à une entité de modèle circulaire. L'un de ces paramètres est une chaîne à plusieurs valeurs et l'autre une masse avec des limites supérieure et inférieure.

1. Ouvrez le document KwrBallBearing1.CATPart.
2. Dans l'arbre de spécifications, sélectionnez l'entité racine, puis la commande **Démarrer->Knowledgeware->Knowledge Advisor** pour ouvrir l'atelier Knowledge Advisor.
3. Dans l'arbre de spécifications, sélectionnez **CircPattern.1**.
4. Cliquez sur l'icône **Explorateur de paramètres**  . La boîte de dialogue suivante s'affiche :



5. Dans la liste **Créer paramètre de type**, sélectionnez le type **Chaîne**, puis dans le champ situé en face ('**Avec**'), sélectionnez Valeurs multiples. Cliquez sur **Créer paramètre de type**.

6. Dans la boîte de dialogue <**Liste de valeurs** :

   a) entrez la chaîne Type1, puis appuyez sur **Entrée**.

   b) entrez la chaîne Type2, puis appuyez sur **Entrée**.

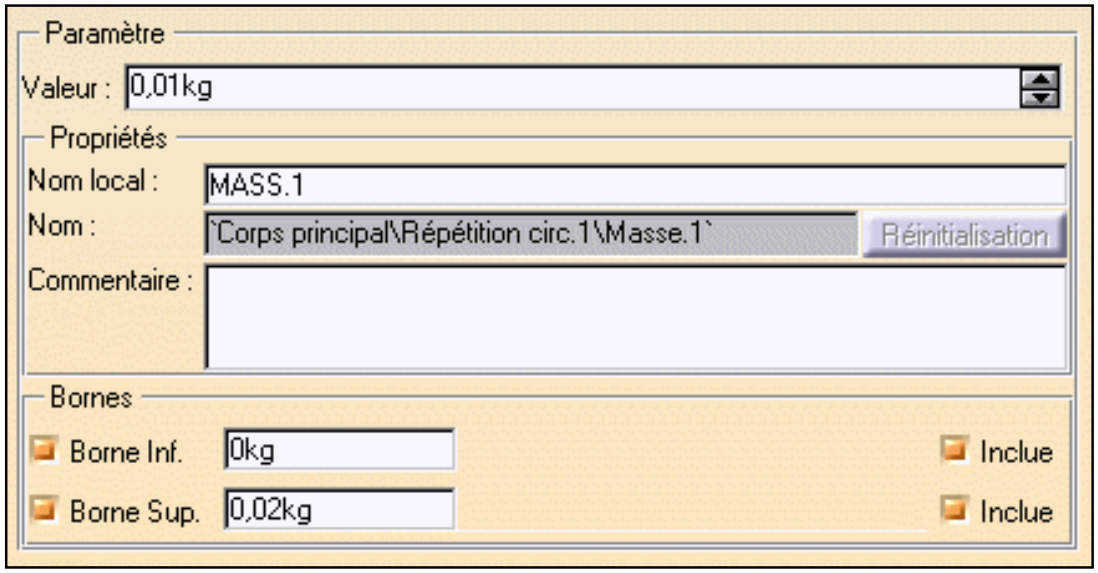
   c) cliquez sur **OK** pour revenir à la boîte de dialogue **Explorateur de paramètres**.

7. Le cas échéant, renommez le paramètre créé dans le champ **Nom local** et ajoutez un commentaire.

8. Dans la liste **Créer paramètre de type**, sélectionnez le type **Masse**, puis dans le

champ situé en face ("Avec"), sélectionnez Simple valeur. Cliquez sur **Créer paramètre de type**. MASSE.1 s'affiche par défaut dans le champ Propriétés et la valeur par défaut 0kg est affectée au paramètre créé.

9. Modifiez ces valeurs comme indiqué dans la figure ci-dessous :



10. Cliquez sur **OK**. Les deux paramètres s'affichent dans l'arbre de spécifications, juste en dessous de l'entité CircPatter.1.

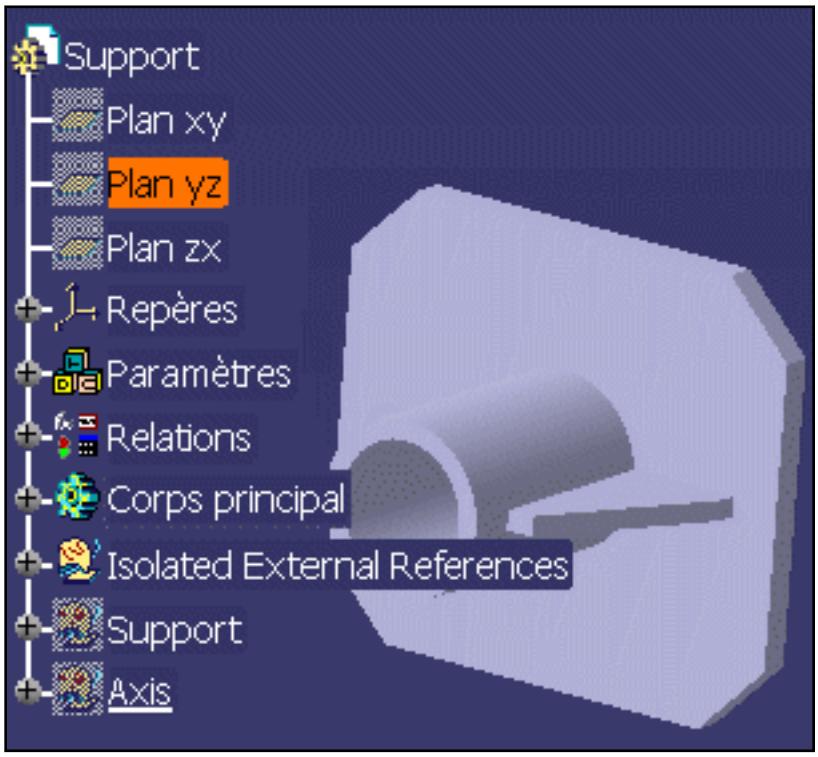
Les paramètres ajoutés à l'aide de l'explorateur de paramètres sont affichés juste en dessous des entités auxquelles ils sont affectés.

Ajout d'un paramètre à une arête

Dans cette tâche, vous apprendrez à ajouter des paramètres à une arête à l'aide de l'icône Explorateur de paramètres.

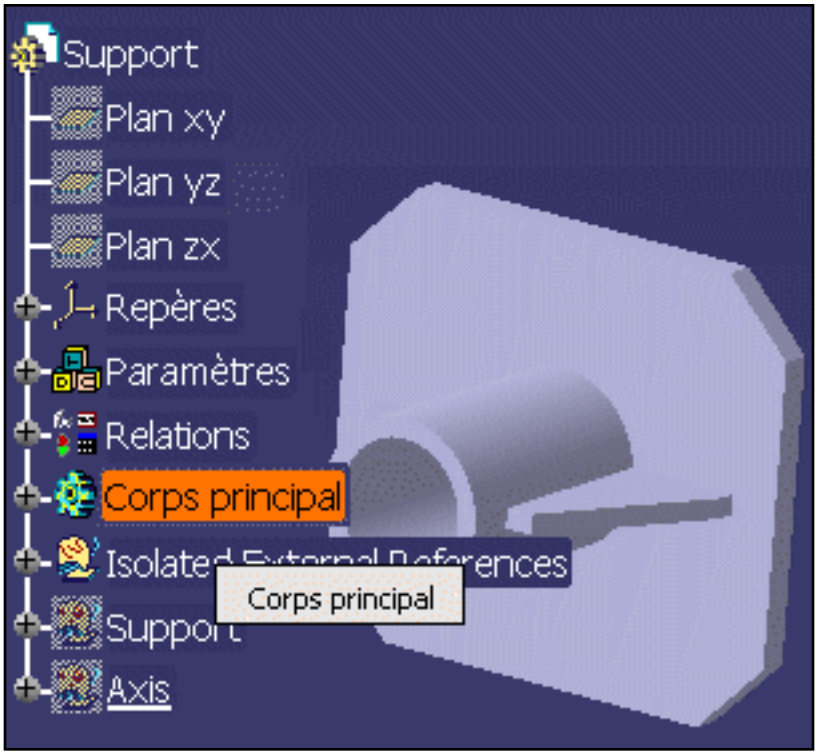
Remarquez que cette nouvelle fonction est conçue pour fonctionner sur des arêtes, des faces et des sommets.

1. Ouvrez le fichier KwrSupport.CATPart. L'image suivante s'affiche.



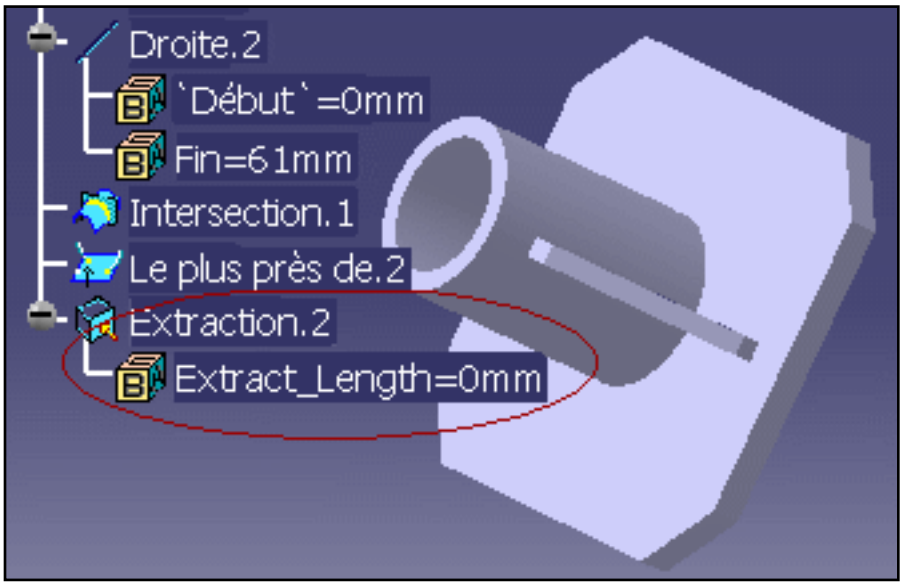
2. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier Knowledge Advisor.

3. Cliquez sur l'icône **Ajout de paramètres sur la géométrie** ( ) et sélectionnez l'arête supérieure de l'élément Pad.2.



La boîte de dialogue Explorateur de paramètres s'ouvre.

4. Dans la liste déroulante **Créer paramètre de type**, sélectionnez le type de paramètre **Longueur**, puis cliquez sur le bouton **Créer paramètre de type**.
5. Dans le champ **Nom local**, entrez le nom du paramètre. Pour les besoins de ce scénario, entrez Extract_Length, puis cliquez sur **OK** pour valider.
   Une nouvelle extraction est créée et le paramètre que vous venez de créer est ajouté à cette entité.



Les paramètres ajoutés à l'aide de l'explorateur de paramètres sont affichés juste en dessous des entités auxquelles ils sont affectés.

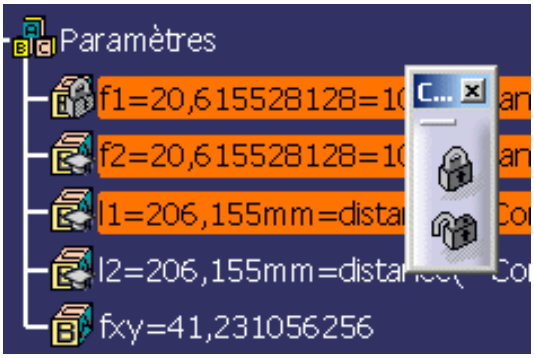
Verrouillage et déverrouillage d'un paramètre

Dans cette tâche, vous apprendrez à verrouiller et déverrouiller des paramètres : deux nouvelles options sont désormais accessibles dans la barre d'outils Knowledge : **Verrouille les paramètres sélectionnés...** et **Déverrouille les paramètres sélectionnés...**.

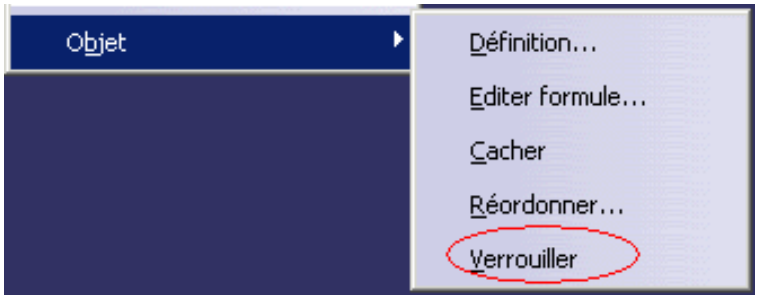
1. Ouvrez le fichier KwrLockingUnlockingParameters.CATPart.
2. Développez le noeud Paramètres et cliquez sur le paramètre f1.
3. Dans la barre d'outils **Knowledge**, cliquez sur l'icône**Verrouille les paramètres sélectionnés...** () pour verrouiller ce paramètre. Le paramètre est verrouillé (un verrou apparaît en regard du paramètre dans l'arbre de spécifications).
4. Pour déverrouiller un paramètre, sélectionnez le paramètre f1 dans l'arbre de spécifications et cliquez sur l'icône **Déverrouille les paramètres sélectionnés...** () dans la barre d'outils Knowledge.

Remarquez les situations suivantes :

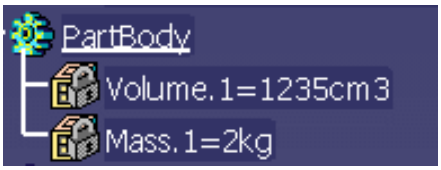
- Il est possible d'utiliser la sélection multiple de paramètres ou de composants avec les options **Verrouille les paramètres sélectionnés...** et **Déverrouille les paramètres sélectionnés...**.



- Le menu contextuel du paramètre peut être utilisé pour le verrouillage ou le déverrouillage du paramètre.



- Les options **Verrouille les paramètres sélectionnés...** et **Déverrouille les paramètres sélectionnés...** sont récursives : si vous sélectionnez un composant, les paramètres situés en amont sont verrouillés ou déverrouillés.



- Il est possible de verrouiller et déverrouiller les paramètres sans représentation 3D. Pour ce faire, procédez comme suit :
  - Ouvrez l'éditeur de formules pour accéder au paramètre.
  - Sélectionnez le paramètre dans la liste des paramètres.
  - Dans le champ **Editer le nom ou la valeur du paramètre sélectionné**, ouvrez le menu contextuel et sélectionnez **Verrouiller/Déverrouiller**. (Cliquez sur le graphique ci-contre pour l'agrandir.)



- Il est possible de verrouiller et déverrouiller des contraintes. Pour ce faire, procédez comme suit :
  - Sélectionnez la contrainte dans l'arbre de spécifications et cliquez sur l'icône **Verrouille les paramètres sélectionnés...** ()/**Déverrouille les paramètres sélectionnés** () ou
  - Cliquez avec le bouton droit de la souris dans l'arbre de spécifications et sélectionnez l'option **Bloquer le paramètre/Débloquer le paramètre**.



Un verrou apparaît en regard de la contrainte dans l'atelier Esquisse. Les contraintes bloquées s'affichent en orange.

Création d'ensembles de paramètres

Dans cette tâche, vous apprendrez à créer des ensembles de paramètres. Vous pouvez créer des ensembles de paramètres sous le noeud Paramètres de l'arbre de spécifications. Cette fonction permet de regrouper des paramètres par catégorie.

1. Dans **Outils->Options...->Infrastructure->Part Infrastructure**, cochez au moins les options Paramètres et Relations de l'onglet Affichage.
2. Ouvrez un document contenant au moins un paramètre ou créez un document et ajoutez-y un paramètre (sans quoi le noeud Paramètres ne s'affichera pas dans l'arbre de spécifications).
3. Cliquez sur l'icône  et sélectionnez le noeud Paramètres dans l'arbre de spécifications. L'ensemble de paramètres Paramètres.1 (ou Paramètres.n) est ajouté à l'arbre de spécifications juste en dessous du noeud Paramètres.
4. Cliquez sur l'icône  pour ajouter un nouveau paramètre dans l'ensemble de paramètres créé. La boîte de dialogue Explorateur de paramètres s'affiche. Dans l'arbre de spécifications, sélectionnez l'ensemble de paramètres auquel vous souhaitez ajouter un paramètre. Le nom de l'ensemble de paramètres s'affiche dans le champ Feature de la boîte de dialogue Explorateur de paramètres.
5. Renseignez les autres champs de la boîte de dialogue Explorateur de paramètres. Le cas échéant, reportez-vous à la section Ajout d'un paramètre à une entité.
6. Une fois le nouveau paramètre défini, cliquez sur OK dans la boîte de dialogue Explorateur de paramètres. Dans l'arbre de spécifications, vous pouvez développer l'entité représentant l'ensemble de paramètres. Un nouveau paramètre a été ajouté sous l'ensemble de paramètres.

Pour réordonner les paramètres appartenant à un ensemble de paramètres, utilisez la commande **Réordonner...** du menu contextuel.

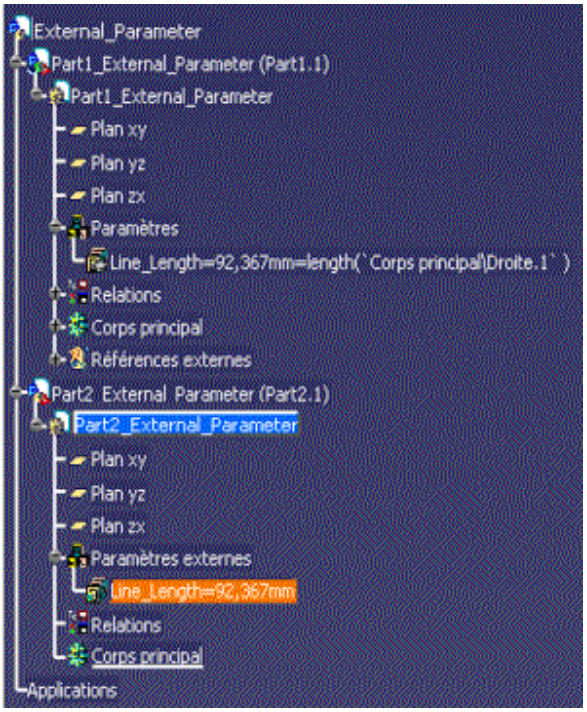
Création d'un paramètre externe

Dans cette tâche, vous apprendrez à créer un paramètre externe.

- Vérifiez que vous avez activé :
  - les options **Paramètres** et **Références externes** (en dessous de la section Arbre de spécifications) dans la boîte de dialogue **Outils -> Options -> Infrastructure -> Part Infrastructure -> Affichage** ,
  - ainsi que les options **Avec formule** et **Avec valeur** (en dessous de la section Vue des paramètres dans l'arbre) dans la boîte de dialogue **Outils -> Options -> Général -> Paramètres et mesure**.
- Ouvrez le fichier DeactivatingExternalParameter.CATProduct.

Notez que les paramètres externes créés au niveau d'un produit ne sont pas intégrés au processus de mise à jour.

1. Développez le document Part1_External_Parameter(Part1.1)\Part1_External_Parameter, cliquez avec le bouton droit de la souris sur le paramètre Line_Length sous le noeud Paramètres dans l'arbre de spécifications, puis sélectionnez la commande **Copier**.
2. Cliquez deux fois sur Part2_External_Parameter (Part2.1)\Part2_External_Parameter, puis cliquez avec le bouton droit de la souris, et sélectionnez la commande **Collage spécial**. Dans la boîte de dialogue Collage spécial, sélectionnez **En tant que résultat avec lien** et cliquez sur **OK** pour valider. Le noeud Paramètres externes est ajouté dans l'arbre de spécifications et le paramètre Line_Length apparaît.



Désactivation et réactivation de paramètres externes

Dans ces tâches, vous apprendrez à désactiver, réactiver et synchroniser des paramètres externes. Dans le scénario ci-après, l'utilisateur :

- Crée un paramètre externe
- Désactive le paramètre externe
- Modifie le paramètre auquel fait référence le paramètre externe
- Réactive le paramètre externe

Jusqu'à présent, les paramètres externes (références externes) pouvaient être :

- synchronisés (si leur valeur était différente de celle du paramètre de référence) ;
- isolés (les paramètres externes deviennent des paramètres simples).

Dorénavant, les utilisateurs peuvent désactiver et réactiver les paramètres externes. Autrement dit, une référence externe, lorsqu'elle est désactivée, ne peut pas être synchronisée avec la valeur à laquelle il est fait référence, mais conserve sa dernière valeur (la valeur n'est plus modifiable par l'utilisateur). Le paramètre externe peut également être réactivé : son comportement natif est alors restauré.

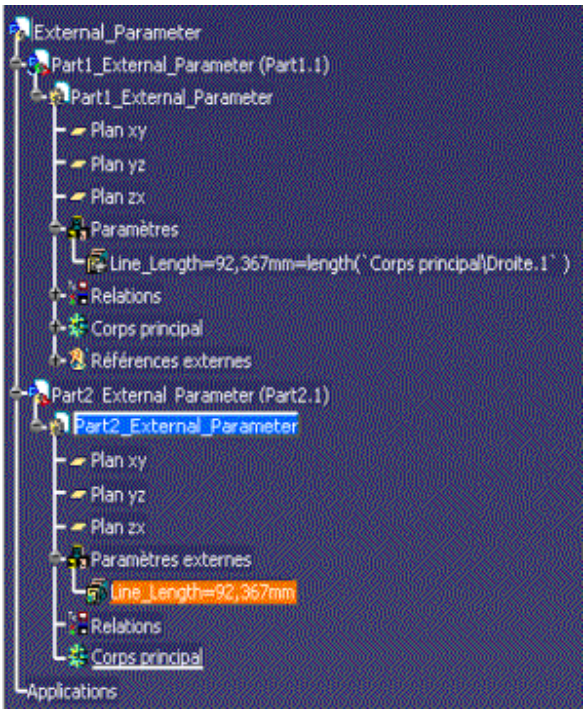
Remarquez les situations suivantes :

- Un paramètre externe désactivé perd simplement sa capacité de synchronisation et conserve sa valeur la plus récente.
- Il est possible de réactiver un paramètre externe désactivé à l'aide de son sous-menu contextuel.
- Il est possible d'accéder aux commandes **Activer**, **Désactiver**, **Synchroniser**, et **Isoler** dans la boîte de dialogue **Liaisons** (**Edition -> Liaisons**.)

- Vérifiez que vous avez activé :
  - les options **Paramètres** et **Références externes** (en dessous de la section Arbre de spécifications) dans la boîte de dialogue **Outils -> Options -> Infrastructure -> Part Infrastructure -> Affichage**
  - ainsi que les options **Avec formule** et **Avec valeur** (en dessous de la section Vue des paramètres dans l'arbre) dans la boîte de dialogue **Outils -> Options -> Général -> Paramètres et mesure**.
- Ouvrez le fichier DeactivatingExternalParameter.CATProduct.

Création du paramètre externe

1. Développez le document Part1_External_Parameter(Part1.1)\Part1_External_Parameter, cliquez avec le bouton droit de la souris sur le paramètre Line_Length sous le noeud Paramètres dans l'arbre de spécifications, puis sélectionnez la commande **Copier**.
2. Cliquez deux fois sur Part2_External_Parameter (Part2.1)\Part2_External_Parameter, puis cliquez avec le bouton droit de la souris, et sélectionnez la commande **Collage spécial**. Dans la boîte de dialogue **Collage spécial**, sélectionnez **En tant que résultat avec lien** et cliquez sur **OK** pour valider. Le noeud Paramètres externes est ajouté dans l'arbre de spécifications et le paramètre Line_Length apparaît.



Désactivation du paramètre externe

3. A l'aide du bouton droit de la souris, cliquez sur ce paramètre externe et sélectionnez la commande **Part2_External_Parameter\External Parameters\Line_Length object -> Désactiver**. Un masque() indiquant que le paramètre est désactivé s'affiche en regard du nom du paramètre.

4. Cliquez deux fois sur Part1_External_Parameter, cliquez deux fois à l'aide du bouton droit de la souris sur Line_Length dans Part1_External_Parameter et sélectionnez la commande **Line_Length object -> Editer formule...**.

Modification du paramètre pointé par le paramètre externe

5. Dans l'éditeur de formules, modifiez la formule pour qu'elle corresponde à la formule suivante : Line_Length=Length(PartBody\Line.1)*2. Lorsque vous avez terminé, cliquez sur **OK**. La valeur du paramètre Line_Length est mise à jour dans Part1_External_Parameter. Notez qu'elle n'est pas mise à jour dans Part2_External_Parameter puisque ce paramètre externe est désactivé.

Réactivation du paramètre externe

6. Revenez au paramètre externe que vous avez créé, cliquez dessus avec le bouton droit de la souris et sélectionnez la commande **Part2_External_Parameter\External Parameters\Line_Length object -> Activer**. Un masque() indiquant que le paramètre est désactivé s'affiche en regard du nom du paramètre.

7. A l'aide du bouton droit de la souris, cliquez sur ce paramètre externe et sélectionnez la commande **Part2_External_Parameter\External Parameters\Line_Length object -> Synchroniser**.Le paramètre est mis à jour et un masque() indiquant que le paramètre est désactivé s'affiche en regard du nom du paramètre.



Utilisation de formules



Cliquez sur l'icône Formule pour indiquer des relations entre les paramètres.

Présentation des formules
Familiarisation avec la boîte de dialogue ***f(x)***
Utilisation du dictionnaire
Création d'une formule
Création de formules basées sur des publications
Spécification d'une mesure dans une formule
Utilisation d'une géométrie pour créer une formule
Référence à des paramètres externes dans une formule
Utilisation de la fonction Equivalence

Présentation des formules

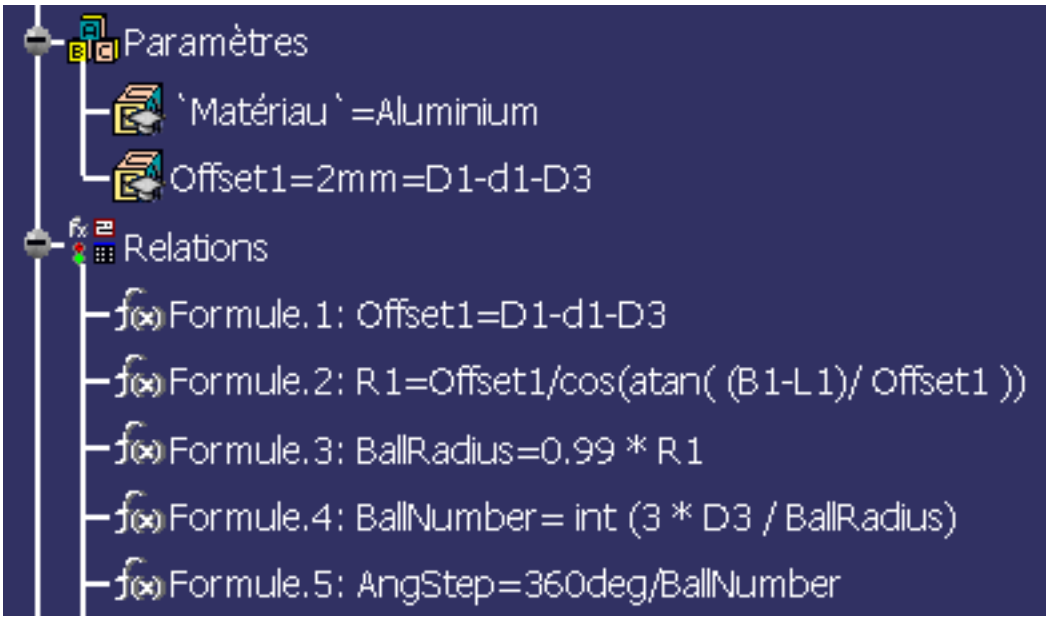
Les formules permettent de définir un paramètre ou de lui imposer des contraintes. Une formule est une relation : la première partie de cette relation désigne le paramètre sujet de la contrainte tandis que la deuxième partie contient l'instruction définissant celle-ci. Une fois créée, la formule peut être gérée comme n'importe quel autre composant, à partir de son menu contextuel. Le langage utilisé dans les formules fait appel à des opérateurs et des fonctions de tous types, grâce auxquels vous pouvez appliquer des opérations sur les paramètres.

Affichage des formules dans l'arbre de spécifications

En tant que relations, les formules peuvent être affichées en aval du noeud Relations si vous avez au préalable coché l'option "Relations" (en dessous de la section "Arbre de spécifications") dans la boîte de dialogue **Outils -> Options -> Infrastructure -> Part Infrastructure -> Affichage**.

D'autre part, les formules peuvent être affichées en aval du noeud Paramètres à condition que vous sélectionniez :

- l'option "Paramètres" (en dessous de la section "Arbre de spécifications") dans la boîte de dialogue **Outils -> Options -> Infrastructure -> Part Infrastructure -> Affichage** ,
- ainsi que la case "Avec formule" (en dessous de la section Vue des paramètres dans l'arbre) dans la boîte de dialogue **Outils -> Options -> Général -> Paramètres et mesure**.



Paramètre Activité

Une formule est un composant auquel est affecté un paramètre appelé *Activité*. La valeur du paramètre activité est un opérateur booléen. Si cette valeur est Vrai, le paramètre ne peut pas être calculé à partir de la formule. Si vous créez une formule pour un paramètre auquel aucune contrainte n'est associée, le paramètre Activité de cette nouvelle formule prend la valeur Vrai par défaut.

Un paramètre peut être rattaché à plusieurs formules de contrainte, mais une seule à la fois peut être active. Avant d'activer une formule donnée pour un paramètre, vous devez désactiver les autres formules définies pour ce paramètre.

Valeur de l'Activité	faux	vrai
Icône Relation dans l'arbre de spécifications		

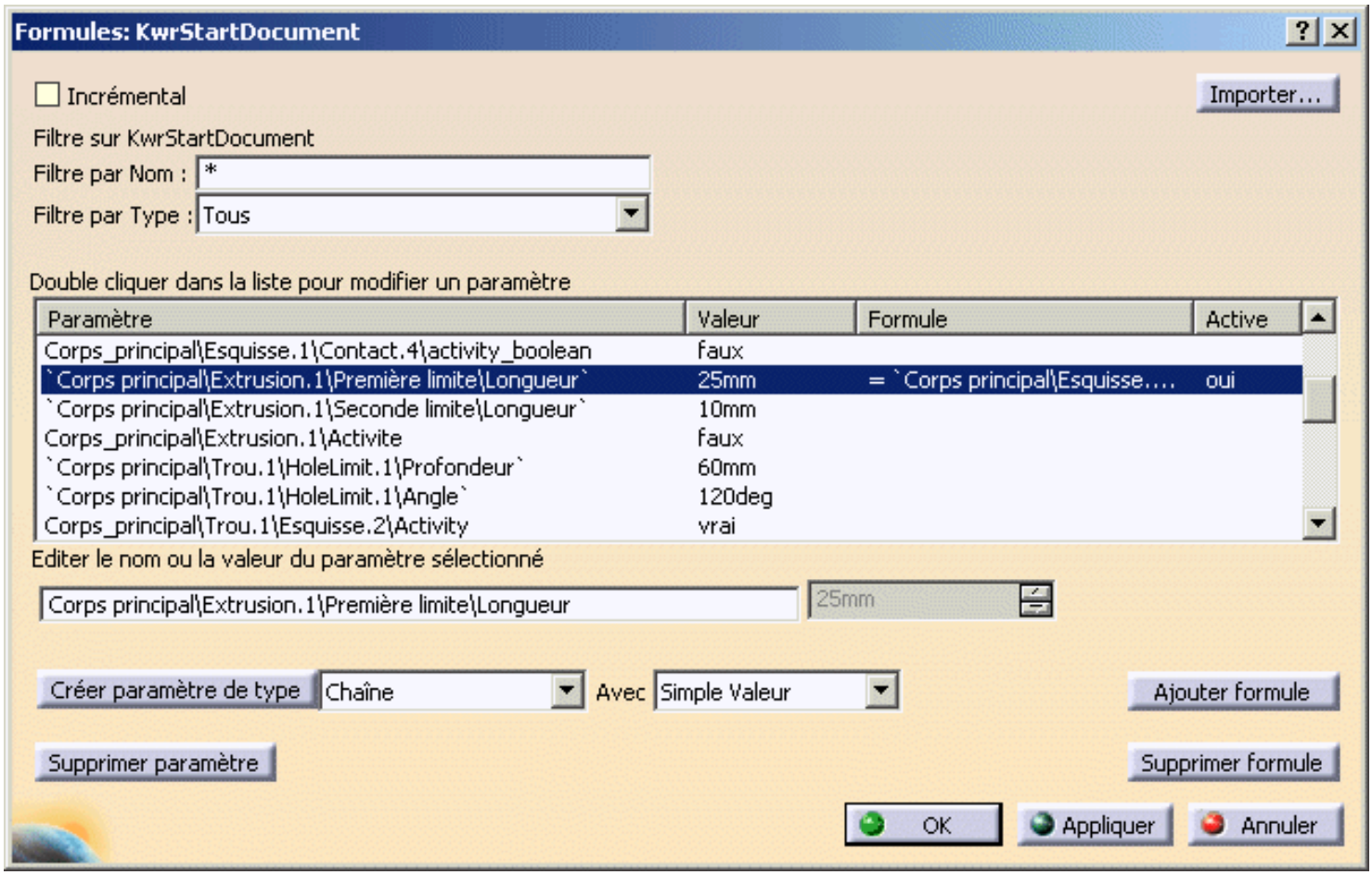
Importation de formules

Comme les paramètres, les formules associées peuvent être importées à partir d'un fichier externe. Pour plus d'informations sur l'importation de formules, reportez-vous aux sections Présentation des paramètres et Importation de paramètres.

Familiarisation avec la boîte de dialogue *f(x)*

La boîte de dialogue *f(x)* s'ouvre lorsque vous cliquez sur l'icône  dans la barre d'outils standard. Elle vous permet :

- d'afficher la liste des paramètres,
- de créer des paramètres et des formules,
- d'importer des fichiers externes.



Liste des paramètres

Comme son nom l'indique, la liste des paramètres affiche les paramètres associés au composant sélectionné dans l'arbre de spécifications ou dans la zone géométrique. En l'absence de sélection d'un composant, tous les paramètres du document sont affichés. Lorsque la boîte de dialogue est ouverte, vous pouvez sélectionner un composant dans l'arbre ou dans la zone géométrique pour obtenir les paramètres s'y rapportant.

Vous pouvez restreindre la liste des paramètres affichés en utilisant les options Filtre par nom et Filtre par type ou en cochant la case Incrémental.

Filtre "Filtre par nom"

Ce filtre vous permet de limiter la liste des paramètres affichés en spécifiant une sous-chaîne. Par exemple, si vous spécifiez *Limit* comme élément de filtrage, seul le paramètre contenant la sous-chaîne Limit seront affichés, par exemple :

PartBody\Pad.1\FirstLimit\Length

PartBody\Pad.1\SecondLimit\Length
PartBody\Trou.1\HoleLimit.1\Depth
PartBody\Trou.1\HoleLimit.1\Angle

Filtre "Filtre par type"

Ce filtre vous permet de limiter la liste des paramètres affichés en spécifiant un type. Si vous sélectionnez Paramètres utilisateurs, seuls les paramètres créés à partir de l'option "Créer paramètre de type" seront affichés. Si vous sélectionnez Paramètres cachés, la liste n'affichera que les paramètres dont le non-affichage a été demandé via la commande "Cacher" du menu contextuel associé au champ de valeur.



La commande "Cacher" n'est disponible que pour les paramètres utilisateur.

Option Incrémental

Lorsque vous sélectionnez un composant dans la zone géométrique ou dans l'arbre de spécifications, seul le premier niveau de composants est affiché dans l'éditeur, en dessous du composant sélectionné. La liste des paramètres représentée ci-dessus répertorie tous les paramètres relatifs aux composants Extrusion.1 et Trou.1. Si vous sélectionnez Extrusion.1 dans l'arbre (avec l'option Incrémental non sélectionnée), vous obtenez la liste suivante :

PartBody\Pad.1\FirstLimit\Length
PartBody\Pad.1\SecondLimit\Length
PartBody\Esquisse.1\Rayon.3\Rayon

Si vous sélectionnez Incrémental, la liste se limite aux paramètres suivants :

PartBody\Pad.1\FirstLimit\Length
PartBody\Pad.1\SecondLimit\Length

Champ "Editer le nom ou la valeur du paramètre sélectionné"

Ce champ affiche le paramètre qui a été sélectionné dans la liste des paramètres. Le champ de valeur, qui se trouve à droite, apparaît grisée lorsque le paramètre est contraint par une formule, une table de paramétrage ou un type quelconque de relations. Si vous cliquez avec le bouton droit de la souris dans ce champ de valeur, vous accédez à un certain nombre de commandes qui vous permettent d'afficher la définition du paramètre.

Bouton "Créer paramètre de type"

Ce bouton permet de créer un paramètre utilisateur et de lui associer, selon le cas, une valeur unique ou plusieurs valeurs.

Bouton Supprimer paramètre

Cette commande ne fonctionne que pour les paramètres utilisateur.

Bouton Ajouter formule

Lorsque vous créez une formule, vous spécifiez qu'un paramètre, quel que soit son type, doit être contraint par une relation. Le bouton **Ajouter formule** permet d'accéder à l'éditeur de formules. Une fois créée, la formule apparaît dans la liste des paramètres, avec son activité.

Pour avoir plus d'informations sur le dictionnaire disponible dans l'éditeur de formules, reportez-vous à la section Utilisation du dictionnaire.

Bouton Supprimer formule

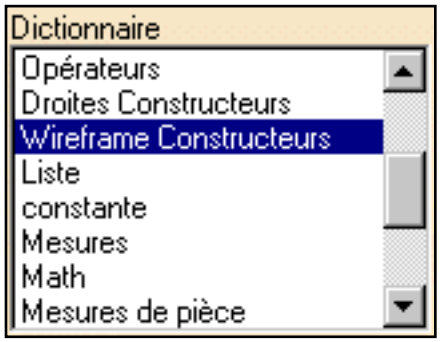
Lorsqu'un paramètre contraint par une formule est sélectionné dans la liste des paramètres, vous pouvez supprimer cette formule en cliquant sur le bouton Supprimer formule.

Bouton Importer

Cette fonction permet d'importer des paramètres et leurs valeurs à partir d'un fichier texte ou d'un fichier Excel (Windows).

Utilisation du dictionnaire

Le dictionnaire vous permet d'accéder aux fonctions, aux opérateurs et aux attributs de composants qui peuvent être utilisés dans les relations.
Vous pouvez y accéder par l'éditeur de formules, l'éditeur de règles, l'éditeur de vérifications et l'éditeur d'action.



Les packages affichés dans la partie gauche de l'explorateur sont ceux que vous avez sélectionnés via l'onglet **Outils->Options ->Général->Paramètres et mesure->Langage**.

Mots clés	Tables de paramétrage	Opérateurs
Points constructeurs	Loi	Fonction de recherche
Droites constructeurs	Cercles constructeurs	String
Directions constructeurs	Fonctions de l'atelier Electrical	Space Analysis
Messages et macros	Attribut	Liste
Mesures	Surfaces constructeurs	Wireframe constructeurs
Mesures de pièces	Plans constructeurs	Opérateurs d'analyse
Objet	Math	

Constantes

Lors de la programmation des règles et des vérifications, *CATIA* peut reconnaître un certain nombre de constantes. Vous pouvez utiliser ces constantes dans une relation à la place des valeurs proprement dites.

- faux - Une des deux valeurs admises pour un opérateur booléen.
- vrai - Une des deux valeurs admises pour un opérateur booléen.
- PI - **3.14159265358979323846 -** Rapport de la circonférence d'un cercle à son diamètre.
- E - Base du logarithme népérien. La valeur de la constante *e* est approximativement **2,718282**.

Mots clés

if...else ... else... if	let	for
inside	while	

if...else ... else...if

Exécute de façon conditionnelle un groupe d'instructions, suivant la valeur d'une expression. Vous pouvez utiliser l'une des syntaxes suivantes :

```
if instructions conditionnelles[else elsestatements ]


ou


if condition
   { instructions }
[else if condition-n
   [ { instructions elseif } ] ] ...
[else
   [ { instructions else } ] ]
```

Vous pouvez utiliser la forme en une ligne (première syntaxe) pour des règles courtes et simple.Toutefois, la la forme en bloc (seconde syntaxe) fournit davantage de structure et de flexibilité que la forme sur une ligne et est généralement plus facile à lire, à gérer et à vérifier.
Les clauses **else** et **else if** sont facultatives.Vous pouvez avoir autant d'instructions **else if** que vous le souhaitez après un bloc **if**, mais aucune ne peut apparaître après la clause **else**.Les instructions associées au bloc **if** peuvent être imbriquées, c'est-à-dire contenues les unes dans les autres.

Vérifications

instruction1 **=>** *instruction2* (**if** *instruction1* **then** *instruction2*)
Affiche un message (si le type est Avertissement ou Information) et devient rouge dans l'arbre de spécifications chaque fois que l'instruction *instruction2* n'est pas valide lorsque l'instruction *instruction1* est remplie.

OK => KO 

KO => KO 

KO => OK 

OK => OK 

Instruction For

Notez que l'instruction For est disponible uniquement pour les scripts Action et Réaction. Pour créer des boucles, procédez comme suit :

Le premier type de boucle est basé sur l'élément d'une liste. Voir la syntaxe ci-contre.

```
For x inside List
{
... Body ...
}
```

où :

- X est un nom de variable (d'un type donné. Il peut représenter un objet ou une valeur.)
- List est un nom de variable de type List ou une expression renvoyant une liste.
- X (comme toute autre variable du langage) peut être utilisé dans le corps. Il contient le Nième élément de la liste.

Le corps est exécuté N fois, où N correspond au nombre d'éléments de la liste.

Le deuxième type de boucle s'exécute jusqu'à ce qu'une expression ait la valeur "false". Voir la syntaxe ci-contre

```
For x while predicate
{
... Body ...
}
```

où :

- X est un nom de variable de type entier. Il est incrémenté à la fin de chaque exécution du corps.
- Le prédicat est une expression booléenne.Le corps est exécuté tant que cette expression a la valeur "true". Cette expression est calculée avant le corps.

Notez que le deuxième type peut générer des boucles infinies.

Instruction While

Cette boucle s'exécute jusqu'à ce qu'une expression ait la valeur "false". Voir la syntaxe ci-contre

où :

- i est un nom de variable de type entier. Il est incrémenté à la fin de chaque exécution du corps.
- X est une variable pour les points.

```
let i = 1
let x(Point)

for i while
i<=parameter.Size()
{
x = parameter.GetItem(i)
if (x.GetAttributeReal("Y") <
0.04)
x.SetAttributeReal("Y",0.04)
}
```

let

Les variables temporaires peuvent être déclarées à l'aide du mot clé**let** . Une variable temporaire n'est pas conservée en tant que paramètre après l'exécution de la vérification ou de la règle.

```
/*Règle créée par CRE 08/23/99*/
let x = 5 mm
if PartBody\Trou.1\Diameter > x
{
PartBody\Trou.1\Activity = false
}
```

Le type doit être indiqué pour les valeurs non numériques :

```
let S(Surface)
S= split (...,...)
```

Les variables temporaires doivent être déclarées au début de la règle, avant toute spécification d'instruction conditionnelle.

```
let S1(Surface)
let S2(Surface)
let S3(Surface)


S1 = Split ...
S2 = ...
S3 = ...
```

Méthodes de table de paramétrage

Méthode CloserSupConfig | Fonction CloserInfConfig | Méthode CloseValueSupInColumn
Méthode CloseValueInfInColumn | Fonction MinInColumn | Méthode MaxInColumn
Méthode LocateInColumn | Fonction CellAsString | Méthode CellAsBoolean
Méthode CellAsReal | Méthode SetCell | Méthode LocateInRow
Sheet.CloserConfig

Méthode CloserSupConfig

S'applique au fichier source d'une table de paramétrage. Renvoie la configuration contenant les plus grandes valeurs inférieures ou égales à celles des arguments définis. Lorsque plusieurs configurations répondent à cette condition, la méthode trie les configurations possibles dans l'ordre des colonnes défini dans la liste des arguments.

Syntaxe

sheet.**CloserSupConfig**(*columnName*: String, *minValue*: Literal, ...): Integer

La méthode **CloserSupConfig** utilise les arguments suivants :

Arguments	***Description***
columnName	A entrer entre guillemets. Au moins un couple *columnNamei*/*minValuei* est nécessaire.
minValue	Obligatoire. Vous devez spécifier les unités.

Exemple

Pour la table de paramétrage suivante :

	SketchRadius(mm)	PadLim1(mm)	PadLim2(mm)
1	120	60	10
2	130	50	30
3	120	60	25
4	140	50	40

L'expression suivante :

Relations\DesignTable1\sheet_name.CloserSupConfig("PadLim1", 60mm, "SketchRadius", 120mm, "PadLim2", 20mm)

renvoie la valeur 3

Méthode CloserInfConfig

S'applique au fichier source d'une table de paramétrage. Renvoie la configuration contenant les plus grandes valeurs inférieures ou égales à celles des arguments définis. Lorsque plusieurs configurations répondent à cette condition, la méthode trie les configurations possibles dans l'ordre des colonnes défini dans la liste des arguments.

Syntaxe

sheet.**CloserInfConfig**(*columnName*: String, *maxValue*: Literal, ...):Integer

La méthode **CloserInfConfig** utilise les arguments suivants :

Arguments	***Description***

columnName	A entrer entre guillemets. Au moins un couple *columnName*/*maxValue* est nécessaire.
maxValue	Obligatoire. Vous devez spécifier les unités.

Exemple

Pour la table de paramétrage suivante :

	SketchRadius(mm)	PadLim1(mm)	PadLim2(mm)
1	120	60	10
2	130	50	30
3	120	60	20
4	140	50	40

L'instruction suivante
Relations\DesignTable1\sheet_name.CloserInfConfig("PadLim1", 60mm, "SketchRadius", 130mm, "PadLim2", 40mm)
renvoie la valeur 3.

Explications

Les valeurs des lignes 1, 2 et 3 sont toutes inférieures ou égales aux valeurs spécifiées dans les arguments de la méthode.

- Dans la mesure où le premier paramètre défini dans la liste des arguments est "PadLim1", la méthode analyse les lignes 1, 2 et 3, et recherche la plus grande valeur de "PadLim1" inférieure ou égale à 60 mm. Deux configurations répondent à cette condition : configuration 1 et configuration 3.
- Dans la mesure où le deuxième paramètre défini est "SketchRadius", la méthode analyse les configurations 1 et 3, et recherche la plus grande valeur de "SketchRadius" inférieure ou égale à 130 mm. La fonction trouve à nouveau deux configurations répondant aux critères.
- Elle analyse ensuite les lignes 1 et 3, et recherche la plus grande valeur de "PadLim2" inférieure ou égale à 40 mm. Le résultat est la ligne 3.



Méthode CloserValueSupInColumn

S'applique au fichier source d'une table de paramétrage. Analyse les valeurs d'une colonne et renvoie la plus grande valeur de cellule qui se rapproche le plus de celle définie. Renvoie la valeur 0 si aucune valeur n'est trouvée ou si les arguments de la méthode ne sont pas définis correctement.

Syntaxe

sheet.**CloserValueSupInColumn(***columnIndex:* Integer, *Value:* Real**)**

La méthode **CloserValueSupInColumn** utilise les deux arguments suivants :

Arguments	***Description***
columnIndex	Obligatoire. Index de la colonne de table. Nombre entier compris entre 1 et n.
Valeur	Obligatoire. Valeur recherchée. Il doit s'agir d'un nombre réel.

Exemple

```
ValueSup=Relations\DesignTable1\sheet_name.CloserValueSupInColumn(1, 80mm)
Message("Closest sup value is # (0.08 est la valeur attendue)", ValueSup)
```



Méthode CloserValueInfInColumn

S'applique au fichier source d'une table de paramétrage. Analyse les valeurs d'une colonne et renvoie la plus petite valeur de cellule qui se rapproche le plus de celle définie. Renvoie la valeur 0 si aucune valeur n'est trouvée ou si les arguments de la méthode ne sont pas définis correctement.

Syntaxe

sheet.**CloserValueInfInColumn**(*columnIndex*: Integer, *value*: Real): Real

La méthode **CloserValueInfInColumn** utilise les deux arguments suivants :

Arguments	***Description***
columnIndex	Obligatoire. Numéro ou index de la colonne de table. Nombre entier compris entre 1 et n.
value	Obligatoire. Valeur recherchée. Il doit s'agir d'un nombre réel.

Exemple

```
Message("La valeur inférieure la plus proche est # ",
Relations\DesignTable1\sheet_name.CloserValueInfInColumn(2,41mm))
```

Méthode MinInColumn

S'applique au fichier source d'une table de paramétrage. Renvoie la plus petite des valeurs de colonne. Renvoie 0 si la colonne définie n'est pas comprise dans l'intervalle autorisé.

Syntaxe

sheet.**MinInColumn**(*columnIndex* : Index): Real

où *columnIndex* est le numéro de la colonne.

Exemple

```
MinimumValue=MinInColumn(3)
Message("La valeur minimale est # (0 est la valeur attendue)", MinimumValue)
/* vous pouvez également utiliser */
Message("La valeur minimale est # (0 est la valeur attendue)", MinInColumn(3))
```

Modèle

KwrProgramDT.CATPart

Méthode MaxInColumn

S'applique au fichier source d'une table de paramétrage. Renvoie la plus grande des valeurs d'une colonne. Renvoie la valeur 0 si la colonne ne contient aucune valeur numérique ou si les arguments de la méthode ne sont pas définis correctement.

Syntaxe

sheet.**MaxInColumn**(*columnIndex*: Integer): Real

Exemple

```
MaximumValue=Relations\DesignTable1\sheet_name.MaxInColumn(1)
Message("Maximum value is # (0.150 est la valeur attendue)", MaximumValue)
```



Méthode LocateInColumn

S'applique au fichier source d'une table de paramétrage. Renvoie l'index de la première ligne qui contient une valeur spécifiée. Renvoie la valeur zéro si aucune valeur n'est trouvée ou si les arguments de la méthode ne sont pas définis correctement.

Syntaxe

sheet.**LocateInColumn**(*columnIndex*: Integer, *value*: Literal) : Integer

La méthode **LocateInColumn** utilise les deux arguments suivants :

Arguments	***Description***
ColumnNumber	Obligatoire. Numéro ou index de la colonne de table. Nombre entier compris entre 1 et n.
Valeur	Obligatoire. Valeur recherchée. Il peut s'agir d'une chaîne ou d'un opérateur booléen.

Exemple

```
Line=Relations\DesignTable1\sheet_name.LocateInColumn(4,11mm)
if (Line == 0)
{
Message("Aucune valeur n'a été trouvée !!!")
}
```



Méthode CellAsString

S'applique au fichier source d'une table de paramétrage. Renvoie le contenu d'une cellule se trouvant dans une colonne. Renvoie une chaîne vide si la cellule est vide ou si les arguments de la méthode ne sont pas définis correctement.

Syntaxe

sheet.**CellAsString(***rowIndex:* Integer**, c***olumnIndex:* Integer**):** String

où *rowIndex* est le numéro de la configuration et**c***olumnIndex* le numéro de la colonne.

Exemple

```
CString=Relations\DesignTable1\sheet_name.CellAsString(1,5)
if (CString == "")
 {
  Message("Aucune valeur n'a été lue !!!")
 }
```



Méthode CellAsBoolean

S'applique au fichier source d'une table de paramétrage. Renvoie le contenu d'une cellule se trouvant dans une colonne destinée aux valeurs booléennes. Renvoie la valeur false si la cellule ne contient aucune valeur booléenne ou si les arguments de la méthode ne sont pas définis correctement.

Syntaxe

sheet.**CellAsBoolean**(*rowIndex*: Integer, *columnIndex*: Integer): Boolean

La méthode **CellAsBoolean** utilise les deux arguments suivants :

Arguments	***Description***
rowIndex	Obligatoire. Numéro de configuration. Nombre entier compris entre 1 et n.
columnIndex	Obligatoire. Index de la colonne de table. Nombre entier compris entre 1 et n.

Exemple

```
Boolean2=Relations\DesignTable1\sheet_name.CellAsBoolean(1,5)
if (Boolean2 <> true)
{
 Message("Erreur !!!")
}
```



Méthode CellAsReal

S'applique au fichier source d'une table de paramétrage. Renvoie le contenu d'une cellule se trouvant dans une colonne destinée aux valeurs réelles. Renvoie la valeur zéro si la cellule ne contient aucune valeur réelle ou si les arguments de la méthode ne sont pas définis correctement.

Syntaxe

sheet.**CellAsReal**(*rowIndex*: Integer, *columnIndex*: Integer): Real

où *rowIndex* est le numéro de configuration (nombre entier compris entre 1 et n) et *columnIndex* le numéro de colonne.



Méthode SetCell

Permet à l'utilisateur d'ajouter une cellule à un emplacement donné dans un fichier Excel ou dans un fichier délimité par des tabulations.

Remarque : L'index doit commencer à 1 pour que la cellule (1,1) soit située dans le coin supérieur gauche.

Syntaxe

sheet.**SetCell(***IndexRow:Integer, IndexColumn:Integer, CellValue:Literal***):** Void



Méthode LocateInRow

S'applique au fichier source d'une table de paramétrage. Renvoie l'index de la première ligne qui contient une valeur spécifiée. Renvoie la valeur zéro si aucune valeur n'est trouvée ou si les arguments de la méthode ne sont pas définis correctement.

Syntaxe

sheet.**LocateInRow**(*rowIndex*: Integer, *value*: Literal) : Integer

La méthode **LocateInRow** utilise les deux arguments suivants :

Arguments	*Description*
RowNumber	Obligatoire. Numéro ou index de la ligne de table. Nombre entier compris entre 1 et n.
Valeur	Obligatoire. Valeur recherchée. Il peut s'agir d'une chaîne ou d'un opérateur booléen.



Sheet.CloserConfig(string columnName1, string sortMethod1, any value1, columnName2, sortMethod2, value2, ...)

Méthode permettant à l'utilisateur de trouver la configuration la plus proche d'une table de paramétrage selon les critères qui mélangent les valeurs supérieur à (ou égal à) et inférieur à (ou égal à),

où :

- Les arguments columnName<index> correspondent aux noms de colonne dans le fichier source de la table de paramétrage.
- Les arguments sortMethod<index> sont des chaînes acceptant uniquement les valeurs suivantes : "<", "<=", ">", ">=", "==", "!="
- Les arguments value<index> peuvent être des valeurs numériques (la magnitude doit correspondre à celle d'une colonne donnée) ou des valeurs de chaîne. S'il s'agit de valeurs de chaîne, la valeur exacte est récupérée des cellules (l'argument sortMethod est utilisé uniquement dans ce cas "==" ou "!=".) En l'absence de l'argument sortmethod "==" ou "!=", "==" .

Exemple :

Sheet.CloserConfig("column1", "<", 10mm, "column2", ">=", 20deg, "column3", "!=", "standard") qui peut être interprété comme suit :
Trouvez la configuration où :

- la valeur de column1 est strictement inférieure à 10mm,
- la valeur de column2 est supérieure ou égale à 20 degrés,
- la valeur de column3 est différente de la chaîne standard.

Remarquez les situations suivantes :

- Si plusieurs configurations sont valides, vous trouvez celle dont les valeurs sont les plus proches possible des "limites". Si plusieurs configurations valides sont détectées, la première est renvoyée.
- Si aucune configuration n'est valide, la valeur renvoyée est 0.

Opérateurs

Opérateurs arithmétiques

+ Opérateur d'addition (permet aussi de concaténer des chaînes)

- Opérateur de soustraction

* Opérateur de multiplication

/ Opérateur de division

( ) Parenthèses (pour regrouper les opérandes dans des expressions)

= Opérateur d'affectation

** Opérateur d'élévation à une puissance

Opérateurs logiques

et Conjonction logique de deux expressions

ou Disjonction logique de deux expressions

Opérateurs de comparaison

<> Différent de

== Egal à

>= Supérieur ou égal à

<= Inférieur ou égal à

< Inférieur à

> Supérieur à

Point Constructeurs

Modèle : KwrPointConstructors

- **point** (*x*: Length, *y*: Length, *z*: Length): Point
  Crée un point à partir de trois coordonnées. Des valeurs ou des noms de paramètre peuvent être utilisés pour entrer les arguments.

Exemples :
Spécification des valeurs :
Geometrical Set.1\Point.1 =
point(10mm,10mm,10mm)
Spécification de noms de paramètre :
Geometrical Set.1\Point.4 =
point(0mm,L3,L1)

- **pointbetween**(*pt1*: Point, *pt2*: Point, *ratio*: Real, *orientation*: Boolean) : Point
  Crée un point entre deux autres points. Si la valeur du quatrième argument est true, le troisième paramètre a pour valeur le rapport entre la distance pt1-nouveau point et la distance pt1-pt2. En revanche, si sa valeur est false, il s'agira du rapport entre la distance pt2-nouveau point et la distance pt1-pt2 (pour créer un point à égale distance de pt1 et de pt2, entrez 0,5).

Exemple :
Geometrical Set.1\Point.5 =
pointbetween(Geometrical Set.1\Point.1, Geometrical Set.1\Point.2, 0.6, true)

- **pointoncurve**(*crv*:Curve, *pt*:Point, *distance*:Length, *orientation*: Boolean) : Point
  Crée un point sur une courbe. Vous devez créer le point à une *distance*curviligne spécifique du point de référence indiqué par le deuxième argument. La valeur booléenne définie par le quatrième argument permet d'inverser la direction dans laquelle le point sera créé. Si le point défini par le deuxième argument n'est pas sur la courbe, la projection de ce point sur la courbe devient le point de référence.

Exemple :
Geometrical Set.1\Point.6 =
pointoncurve(Geometrical Set.1\Spline.1, Geometrical Set.1\Point.5, 5mm, true)

- **pointoncurveRatio**(*crv*:Curve, *pt*:Point, *ratio*:Real, *orientation*: Boolean) : Point
  Crée un point sur une courbe. La position du point à créer est déterminée par le nombre réel défini par le troisième argument. Ce nombre réel est le rapport entre la distance [point à créer -> point de référence] et la distance [point à créer -> extrémité de la courbe]. La valeur booléenne définie par le quatrième argument permet d'inverser la direction dans laquelle le point sera créé. Si le point défini par le deuxième argument n'est pas sur la courbe, la projection de ce point sur la courbe devient le point de référence.

Exemple :
Geometrical Set.1\Point.7 =
pointoncurveRatio(Geometrical Set.1\Spline.1,Geometrical Set.1\Point.3, 0.4,true)

- **pointonplane**(*pln*:Plane, *pt*:Point, *dx*:Length, *dy*:Length): Point
  Crée un point sur un plan. La position du point à créer sur le plan est déterminée par les coordonnées (système H,V) transmises par les troisième et quatrième arguments. Ces valeurs sont spécifiées par rapport au point de référence transmis par le deuxième argument.

Exemple :
```
Geometrical Set.1\Point.8   =
pointonplane(Geometrical Set.1\Plane.1,Geometrical Set.1\Point.1, 10mm,10mm)
```

- **pointonsurface**(*sur*:Surface, *Pt*:Point, *Dir*:Direction, *dist*:Length): Point
  Crée un point sur la surface. La position du point à créer sur la surface est déterminée par la distance (quatrième argument) entre ce point et celui de référence (deuxième argument) le long d'une direction (troisième argument).

Exemple :
```
Geometrical Set.1\Point.9   =
pointonsurface(Geometrical Set.1\Extrude.1,Geometrical Set.1\Point.3,
direction(Geometrical Set.1\Line.1),10mm)
```

- **center**(circle): Point
  Crée un point à partir d'un cercle. Le cercle peut être de n'importe quel type (cercle GSM ou esquisse). Le point créé est le centre du cercle.

Exemple :
```
Geometrical Set.1\Point.10   =
circle(Geometrical Set.1\Circle.1)
```

- **pointtangent**(curve,direction): Point
  Crée un point de tangence entre une courbe et une direction.

Exemple :
```
Geometrical Set.1\Point.11   =
pointtangent( Geometrical Set.1\Spline.1, direction(`yz plane`))
```

- **centerofgravity**(Body): Point
  Crée le centre de gravité d'un solide (c'est-à-dire une entité de type PartBody).

Exemple :
```
Geometrical Set.1\Point.12   =
centerofgravity(PartBody)
```

- **curvaturecenter**(crv: Curve, pt: Point): Point
  Crée le centre de courbure d'une courbe pour un point donné.

Exemple :
```
Geometrical Set.1\Point.13   =
curvaturecenter(Geometrical Set.1\Circle.1, Geometrical Set.1\Point.6)
```

- **extremum**(Curve, Direction, Boolean, Direction, Boolean, Direction, Boolean)
  Construit un point extremum. Les entrées sont une courbe, 3 directions et 3 valeurs booléennes.

Exemple :

```
Geometrical Set.1\Point.2=
extremum(`Geometrical Set.1\Circle.1` ,direction(`xy plane` ) ,FALSE,direction(`xy plane`
),TRUE,direction(`xy plane` ),TRUE)
```

- **extremum**(Surface, Direction, Boolean, Direction, Boolean, Direction, Boolean)
  Construit un extremum. Les entrées sont une surface, 3 directions et 3 valeurs booléennes.

- **extremum**(Solid, Direction, Boolean, Direction, Boolean, Direction, Boolean)
  Construit un extremum. Les entrées sont une solide, 3 directions et 3 valeurs booléennes.

Messages et macros

LaunchMacroFromDoc Fonction	Question Fonction
LaunchMacroFromFile Fonction	VBScriptRun
Message Fonction	

LaunchMacroFromDoc Fonction

Exécute une macro enregistrée dans un document à partir d'une règle.
Une macro est enregistrée dans un document lorsque vous ne spécifiez aucun fichier externe avant de l'enregistrer.

Avertissement : il incombe à l'utilisateur de vérifier que la macro qui est exécutée ne va pas entraîner une boucle infinie ou conduire à un crash du système.

Syntaxe

LaunchMacroFromDoc(*nom_macro*)

Exemple

LaunchMacroFromDoc("Macro1")

Fonction Question

Affiche un message dans une boîte de dialogue, attend que l'utilisateur clique sur un bouton et renvoie une valeur indiquant quel bouton l'utilisateur a actionné (vrai pour Oui et faux pour Non).

Syntaxe

Question(String [# String1 # String2 ..., Param1Name, Param2Name, ...]): Boolean

La fonction **Question** requiert un argument obligatoire et accepte plusieurs arguments facultatifs selon que les valeurs des paramètres doivent apparaître ou non dans le message.

Arguments	***Description***

Chaîne	Obligatoire. Chaîne à faire apparaître dans la boîte de dialogue (à entrer entre guillemets).
# String1, Param1Name...	Facultatif. Lorsque les valeurs des paramètres doivent apparaître dans le message, les arguments doivent être définis de la manière suivante : • une chaîne entre guillemets avec le symbole # pour chaque valeur de paramètre à faire apparaître ; • autant d'instructions [, parameter name] que de valeurs de paramètre déclarées par un symbole "#" dans le message.

Utilisez le symbole "|" pour insérer un retour chariot dans un message.

Exemple

```
Boolean2 =
Question("SketchRadius is # | Voulez-vous modifier cette valeur ?",
Corps_Principal\Esquisse.1\Rayon.3\Radius )
```

Notez que vous pouvez utiliser la fonction Question avec la fonction BuildMessageNLS pour l'affichage des questions dans votre langue. Pour mettre en oeuvre cette fonction, utilisez la syntaxe suivante :

question(BuildMessageNLS ("x","xx",a,b))

- x désigne le nom du fichier CATXXX.CATNls dans lequel se trouvera le message NLS (il s'agit du nom du fichier CATXXX sans l'extension CATNls).
- xx désigne le nom clé utilisé dans ce catalogue.
- a et b désignent les arguments (valeurs qui seront remplacées dans le message).

Fonction LaunchMacrofromFile

Exécute un fichier CATScript de macro sur la base d'une règle.

Avertissement : il incombe à l'utilisateur de vérifier que la macro qui est exécutée ne va pas entraîner une boucle infinie ou conduire à un crash du système.

Syntaxe

```
LaunchMacroFromFile("MacroName.CATScript" )
```

Exemple

```
LaunchMacroFromFile("Macro1.CATScript")
```

Run Method

Exécute une macro avec des arguments.

Avertissement : il incombe à l'utilisateur de vérifier que la macro qui est exécutée ne va pas entraîner une boucle infinie ou conduire à un crash du système.

Syntaxe

VB Script.**Run(***valueOrFeature:ObjectType,...***):** Void

où *valueOrFeature* est le nom de l'argument de la macro. Il peut exister plusieurs arguments.

Exemple

La macro VB Script.1 doit être créée avant de créer la règle suivante :

```
if PartBody\Pad.1.HasAttribute("String.1") == true
`VB Script.1` .Run(PartBody\Pad.1 .GetAttributeString("String.1"),PartBody\Pad.1.Name() )
```

Modèle

KwrObject.CATPart

Fonction Message

Affiche un message dans une fenêtre d'informations. Le message peut comporter une ou plusieurs valeurs de paramètre.

Syntaxe

```
Message(String [# String1  # String2 ..., Param1Name, Param2Name, ...] ) : Void
```

La fonction **Message** requiert un argument obligatoire et accepte plusieurs arguments facultatifs selon que les valeurs des paramètres doivent apparaître ou non dans le message.

Arguments	***Description***
Chaîne	Obligatoire. Chaîne à faire apparaître dans la fenêtre d'informations (à entrer entre guillemets).

# String1, Param1Name...	Facultatif. Lorsque les valeurs des paramètres doivent apparaître dans le message, les arguments doivent être définis de la manière suivante : • une chaîne entre guillemets avec le symbole # pour chaque valeur de paramètre à faire apparaître ; • autant d'instructions [, parameter name] que de valeurs de paramètre déclarées par un symbole "#" dans le message.

Utilisez le symbole "|" pour insérer un retour chariot dans un message.

Exemple 1

```
Message("Le rayon externe est : # | Le rayon interne est : #",
PartBody\Esquisse.1\Rayon.3\Radius,
PartBody\Trou.1\Diameter)
```

Exemple 2

Cette fonction peut être utilisée avec la fonction buildMessageNLS.

```
Message (BuildMessageNLS("KwrCATCatalog.CATNls","Zero"))
```

où x,y,z sont des paramètres.



Notez que vous pouvez utiliser la fonction Message avec la fonction BuildMessageNLS pour l'affichage des questions dans votre langue. Pour mettre en oeuvre cette fonction, utilisez la syntaxe suivante :

Message(BuildMessageNLS ("x","xx",a,b))

- x désigne le nom du fichier CATXXX.CATNls dans lequel se trouvera le message NLS (il s'agit du nom du fichier CATXXX sans l'extension CATNls).
- xx désigne le nom clé utilisé dans ce catalogue.
- a et b désignent les arguments (valeurs qui seront remplacées dans le message).

Méthode Evaluate

Permet de calculer une loi Knowledge Advisor ou Generative Shape Design et d'utiliser les données résultantes dans une autre loi.

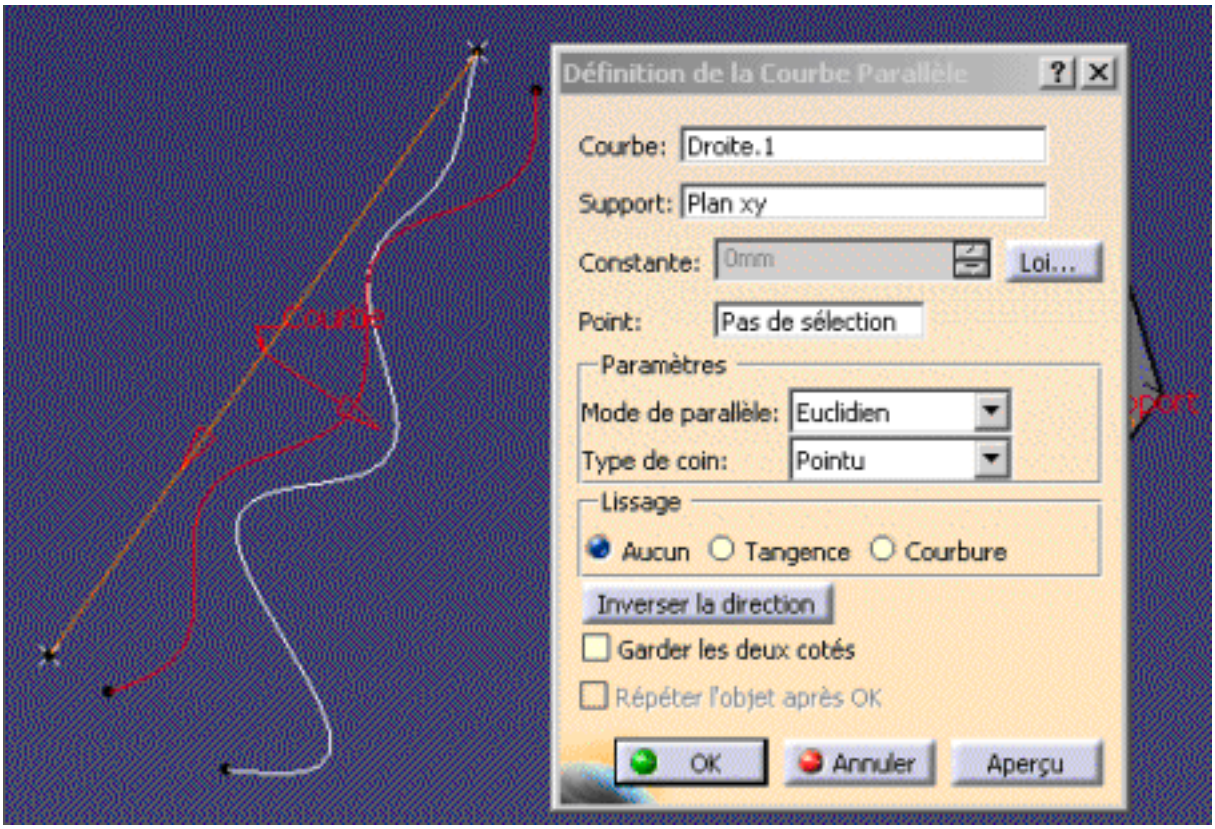
Syntaxe

law.**Evaluate(***Real***):** Real

où l'argument correspond au paramètre auquel la loi est appliquée.

Exemple

1. Créez une droite Generative Shape Design.
2. Pour créer une première loi, cliquez sur l'icône  dans la barre d'outils standard.
3. Dans l'éditeur de lois, créez deux paramètres formels réels.
4. Dans l'éditeur, entrez la loi (Law.1) suivante :
   FormalReal.1 = 5*sin(5*PI*1rad*FormalReal.2)+ 10
5. Cliquez sur OK pour ajouter la loi au document.
6. Répétez la même opération et entrez la deuxième loi (Law.2) suivante :
   FormalReal.1 = 3* FormalReal.2*Relations\Law.1.Evaluate(FormalReal.2)
7. Dans l'atelier Generative Shape Design, créez une droite parallèle à celle créée à l'étape 1. Entrez la loi indiquée ci-dessus dans le champ Décalage.



Modèle

KwrObject.CATPart

Ligne Constructeurs

Modèle : KwrLineConstructors

- **line**(*Point*, *Point*): Line
  Crée une droite à partir de deux points.

Exemple :
```
Geometrical Set.1\Line_Point_Point   =
line( `Geometrical Set.1\Point.1` , `Geometrical Set.1\Point.2` )
```

- **line**(*pt*: Point, *dir*: Direction, *start*: Length, *end*: Length, *orientation*: Boolean) : Line
  Crée une droite passant par un point et parallèle à une direction.
  Les troisième et quatrième arguments sont utilisés pour spécifier le point de début et le point final.
  Le dernier argument permet d'inverser la direction de la droite.

Exemple :
```
Geometrical Set.1\Line.13   =
line( `Geometrical Set.1\Point.2` , direction( `zx plane` ) , 0mm , 20mm , false)
```

- **lineangle**(*crv*: Curve, *sur*: Surface, *pt*: Point, *geodesic*: Boolean, *start*: Length, *end*: Length, *angle*: Angle, *orientation*: Boolean) : Line
  Crée une droite passant par un point, tangente à une surface et effectuant un angle donné avec un courbe. Lorsque l'argument géodésique a la valeur true, une droite géodésique est créée (projetée) sur la surface.

Exemple :
```
Geometrical Set.1\Line.1   =
lineangle( Geometrical Set.1\Spline.1 , Geometrical Set.1\Extrude.1 , Geometrical Set.1\Point.4 ,
false, 0mm , 50mm , 80deg , false)
```

- **linetangent**(*crv*: Curve, *pt*:Point, *start*:Length, *end*:Length, *orientation*:Boolean) : Line
  Crée une droite tangente à une courbe en un point donné.

Exemple :
```
Geometrical Set.1\Line.11   =
linetangent( `Geometrical Set.1\Spline.1` , `Geometrical Set.1\Point.6` ,0mm , 30mm , true )
```

- **linenormal**(*sur*:Surface, *pt*:Point, *start*:Length, *end*:Length, *orientation*:Boolean) : Line
  Crée une droite normale par rapport à une surface en un point donné.

Exemple :
```
linenormal=
linenormal(`Geometrical Set.1\Extrude.1` ,`Geometrical Set.1\Point.2` ,10mm,16mm,true)
```

- **mainnormal**(*crv*: Curve, *pt*: Point) : Line
  Crée une droite normale par rapport à une courbe en un point donné.
  La droite est créée dans le plan contenant le vecteur de tangence.

Exemple :
```
Geometrical Set.1\Line.10=
mainnormal( `Geometrical Set.1\Spline.1` , `Geometrical Set.1\Point.6` )
```

- **binormal**(*crv*: Curve, *pt*: Point) : Line
  Crée une droite normale par rapport à une courbe en un point donné.
  La droite est créée dans le plan orthogonal par rapport au vecteur de tangence.

Exemple :
```
Geometrical Set.1\Line.8=
binormal( `Geometrical Set.1\Spline.1` , `Geometrical Set.1\Point.6` )
```

- **InertiaAxis**(rank: Integer, Body, ...):Line
  Permet de déterminer l'axe d'inertie d'un corps.

Exemple :
```
Geometrical Set.1\Line.9=
inertiaAxis( 1 , `PartBody` )
```

Cercles Constructeurs

Modèle : KwrCircleConstructors.CATPart

- **circleCtrRadius** (*center*: Point, *support*: Surface, *radius*: Length, *limits*: Integer, *start*: Angle, *end*: Angle): Circle

Crée un arc de cercle à partir du centre et du rayon. Si l'argument 4 a la valeur 0, les arguments 5 et 6 sont pris en compte. Sinon, un cercle est créé.

- **circleCtrPt**(*center*: Point, *point*: Point, *support*: Surface, *radius*: Length, *limits*: Integer, *start*: Angle, *end*: Angle): Circle

Crée un arc de cercle à partir du centre et d'un autre point situé sur le cercle. Si l'argument 4 a la valeur 0, les arguments 5 et 6 sont pris en compte. Sinon, un cercle est créé.

- **circle2PtsRadius**(*point1*: Point, *point2*: Point, *support*: Surface, *radius*: Length, *orientation*: Boolean, *limits*: Integer): Circle

Crée un arc de cercle. Les points spécifiés dans les arguments 1 et 2 sont situés sur l'arc à créer et définissent les limites de l'arc lorsque l'entier spécifié dans l'argument 6 est 0. Lorsque 0 est spécifié dans l'argument 6, la modification de la valeur booléenne de l'argument 5 permet d'afficher l'arc alternatif.

- **Circle3Pts** (pt1: Point, pt2: Point, pt3: Point, Limits: Integer) : Circle

Crée un ou plusieurs arcs de cercle passant par trois points. Lorsque le nombre indiqué par l'argument 4 est 0, les premier et troisième points définissent les limites de l'arc. Lorsque le nombre indiqué par l'argument 4 est 1, le cercle complet est défini. Lorsque le nombre indiqué par l'argument 4 est 2, le cercle direct est défini. Lorsque le nombre indiqué par l'argument 4 est 3, le cercle complémentaire est défini.

- **circleBitgtRadius**(*crv1*:Curve, *crv2*:Curve, *support*: Surface, *radius*: Length, *orientation1*: Boolean, *orientation2*: Boolean, *Limits*: Integer) : Circle

Crée un ou plusieurs arcs de cercle tangents aux deux courbes. Lorsque le nombre indiqué par l'argument 7 est 0, les points de tangence définissent les limites de l'arc. La modification de la valeur de l'argument *orientation1* vous permet d'inverser l'orientation de l'arc par rapport à la courbe *crv1* (il se peut qu'il n'existe aucune solution). La modification de la valeur de l'argument *orientation2* vous permet d'inverser l'orientation de l'arc par rapport à la courbe *crv2*.

- **circleBitgtPoint**(crv1:Curve, crv2:Curve, pt:Point , support: Surface, orientation1: Boolean, orientation2: Boolean, Limits: Integer) : Circle

Crée un ou plusieurs arcs de cercle tangents à deux courbes et passant par un point de la seconde courbe. Lorsque le nombre indiqué par l'argument 7 est 0, les points de tangence définissent les limites de l'arc. Si vous modifiez la valeur de l'argument orientation1, vous inversez l'orientation de l'arc par rapport à la courbe crv1 (il se peut qu'il n'existe aucune solution). Si vous modifiez la valeur de l'argument orientation2, vous inversez l'orientation de l'arc par rapport à la courbe crv2.

- **circleBitgtradius**(curve: Curve, point: Point, support; Surface, radius: Length, orientation1: Boolean, orientation2: Boolean, limits: Integer : Circle

Crée un ou plusieurs arcs de cercle tangents aux deux courbes.

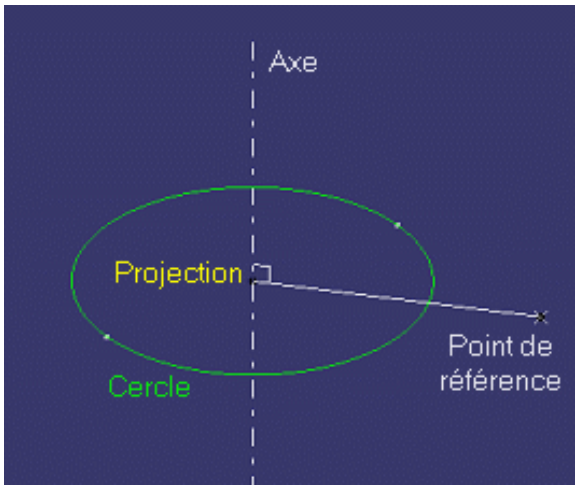
- **circleTritgt**(*curv11*:Curve, *curve2*:Curve, *curve3*:Curve, *support*: Surface, *orientation1*: Boolean, *orientation2*: Boolean, *orientation3*: Boolean, *Limits*: Integer) : Circle

Crée un ou plusieurs arcs de cercle tangents aux trois courbes. Lorsque le nombre indiqué par l'argument 8 est 0, les points de tangence définissent les limites de l'arc. Si vous modifiez la valeur de l'argument *orientation*, vous inversez l'orientation de l'arc par rapport à la courbe ayant le même ordre dans la définition des arguments (l'argument *orientation1* doit être associé à la courbe*crv1*).

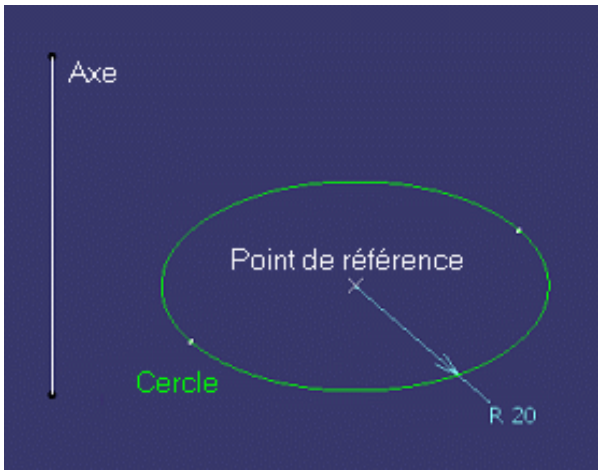
- **circleCtrAxis**(axis: Line, point: Point, radius: Length, start: Angle, end: Angle, projectionmode: Boolean , limits: Integer)

Crée un cercle à l'aide d'un point et d'un axe/d'une droite comme éléments d'entrée, qui correspond au type de cercle "Centre - Axe" dans l'interface utilisateur.

- axix : cercle créé sur un plan perpendiculaire à cet axe.
- point : élément utilisé pour le calcul du centre.
- radius : rayon du cercle créé.
- start : angle de début du cercle. Utilisé uniquement si la valeur des limites est égale à 0.
- end : angle de fin du cercle. Utilisé uniquement si la valeur des limites est égale à 0.
- projectionmode : ProjectionMode = True indique que le point spécifié sera projeté sur l'axe/la droite et sera utilisé comme centre du cercle ; projectionMode = False indique que le point sera le centre du cercle.
- limits : type de limitation du cercle. 0 pour un arc de cercle, 1 pour un cercle complet.



"Point et axe" avec projection

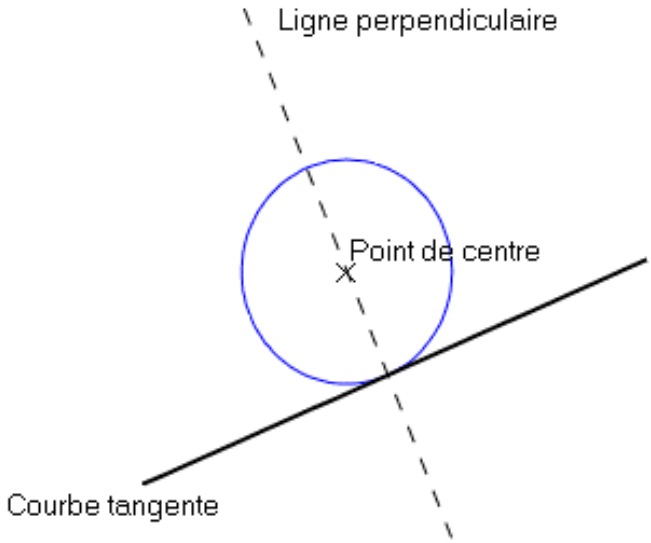


"Point et axe" sans projection

- **circleCtrtgt**(point: Point, curve: Curve, curve orientation: Boolean, tangent orientation: Boolean, support: Surface)

Crées un cercle à l'aide d'un centre, d'une courbe et d'un support, qui correspond au type de cercle "Centre - Tangente" dans l'interface utilisateur.Des orientations sont nécessaires pour permettre la sélection du cercle correct en cas de solutions multiples.

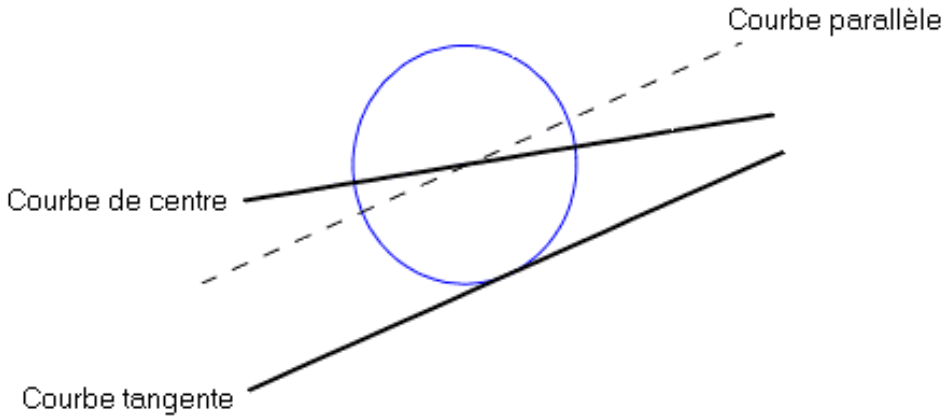
- point : centre du cercle
- curve : courbe avec laquelle le cercle créé sera tangent.
- curve orientation : orientation de la courbe en tangence pour le calcul du cercle.
- tangent orientation : orientation en tangence de la courbe tangente pour le calcul du cercle.
- support : surface du support.



- **circleCtrtgt**(curve1: Curve, curve2: Curve, curve2 orientation: Boolean, curve2 tgt orientation: Boolean, support: Surface, radius: Length, curve1 orientation: Boolean, curve1 tgt orientation: Boolean)

Sert à créer un cercle à l'aide d'un centre, d'une courbe et d'un support, qui correspond au type de cercle "Centre - Tangente" dans l'interface utilisateur. Des orientations sont nécessaires pour permettre la sélection du cercle correct en cas de solutions multiples.

- curve1 : élément central sur lequel repose le centre du cercle. Il s'agit d'une courbe.
- curve2 : courbe avec laquelle le cercle créé sera tangent.
- curve2 orientation : orientation de la courbe en tangence pour le calcul du cercle.
- curve2 tgt orientation : orientation en tangence de la courbe tangente pour le calcul du cercle.
- support : surface du support.
- radius : rayon du cercle créé.
- curve1 orientation : orientation de l'élément central pour le calcul du cercle.
- curve1 tgt orientation : orientation en tangence de l'élément central pour le calcul du cercle.


Courbe parallèle
Courbe de centre
Courbe tangente

Fonction de recherche

Query

Query()
Fonction utilisée pour rechercher les entités situées sous l'entité à laquelle cette fonction s'applique. Elle vérifie l'expression indiquée et ajoute ces entités à la liste.
Dans l'exemple suivant, le résultat de la recherche renvoie les trous de l'entité PartBody dont le diamètre est supérieur à 50 mm.
Exemple : List.1=PartBody.Query("Hole","x.Diameter>50mm")
où :

- List.1 est le nom de la liste concernée par le calcul.
- PartBody est le corps sur lequel portera la recherche.
- Hole est le type de l'entité recherchée.
- x.Diameter>50mm est l'expression.



Dans un contexte d'assemblage, si l'utilisateur exécute une requête sur un type provenant du type Produit ou d'un type affecté à un produit (comme des types électriques, par exemple), la requête renvoie plus d'éléments que prévu puisque toutes les instances de produit (même celles qui ne sont pas visibles) sont renvoyées, ainsi que leurs références.

Chaîne

Méthode Search	Méthode Extract	Méthode Length
Méthode ToString	Méthode ToReal	Méthode ReplaceSubText
Méthode ToUpper	Méthode ToLower	BuildMessageNLS

Méthode Search

Recherche la première occurrence d'une sous-chaîne dans un paramètre de type chaîne. Renvoie l'index du début de la sous-chaîne. Renvoie la valeur -1 si la sous-chaîne définie est introuvable.

Syntaxe

string.**Search**(*StringToBeSearchedFor*)**:** *Integer*

où *StringToBeSearchedFor* est le nom du paramètre de sous-chaîne.

Exemple

```
Responsible = Cilas Evans = ...
SEvans = Evans
I1Search = 6 = Responsible.Search(SEvans)
```

Modèle

KwrString.CATPart

Méthode Extract

Renvoie la sous-chaîne qui commence à une position précise et avec une longueur définie.

Syntaxe

string.**Extract**(*StartIndex*: Integer, *Length*: Integer) : String

où *StartIndex* et *Length* sont respectivement l'index du premier caractère de la sous-chaîne et la longueur de cette sous-chaîne.

Exemple

```
Message (Responsible.Extract(2,2))
```

Modèle

KwrString.CATPart

Méthode Length

S'applique à un paramètre de type chaîne. Renvoie la longueur de la chaîne.

Syntaxe

string.**Length(**Integer

Modèle

KwrString.CATPart

Méthode ToString

Convertit un nombre entier en une chaîne.

Syntaxe

ToString(*Real*) : String

Modèle

KwrString.CATPart

Méthode ToReal

Convertit une chaîne en un nombre réel.

Syntaxe

String.ToReal(*Real***): Real**

Méthode ReplaceSubText

Remplace une sous-chaîne par une autre dans une chaîne de caractères.

Syntaxe

ReplaceSubText(*InputString:* String, *SubStringToBeReplaced:* String, *ReplacingSubString:* String**):** String

Les arguments 2 et 3 peuvent être spécifiés avec les noms des paramètres correspondants ou avec la chaîne elle-même entre guillemets.

Modèle

KwrString.CATPart

Méthode ToUpper

Remplace toutes les minuscules d'une chaîne par des majuscules.

Syntaxe

ToUpper(*StringTobeConverted***:** *String***):** String

où *StringTobeConverted* est le nom d'un paramètre de type chaîne.

Modèle

KwrString.CATPart

Méthode ToLower

Remplace toutes les majuscules d'une chaîne par des minuscules.

Syntaxe

ToLower(*StringTobeConverted***: *String*):** String

où *StringTobeConverted* est le nom d'un paramètre de type chaîne.

Modèle

KwrString.CATPart

BuildMessageNLS

Permet à l'utilisateur d'envoyer des messages ou de poser des questions grâce aux fonctions de message et de question dans la langue de son choix. La fonction BuildMessageNLS peut créer un message NLS (un message dans une langue donnée) en le détectant dans un fichier CATXXX.CATNls.

Notez que cette fonction est pratique lorsqu'elle est utilisée conjointement aux fonctions de message et de question. Pour en savoir plus sur ces 2 fonctions, reportez-vous à la documentation Knowledge Advisor.

Syntaxe

BuildMessageNLS(*MessageCatalog:String, MessageKey: String, argument: Literal, ...*):String

où :

- *MessageCatalog:String* est le nom du fichier CATXXX.CATNls dans lequel se trouve le message NLS (il s'agit du nom CATXXX sans l'extension CATNls).
- *MessageKey: String* est le nom de la clé dans ce catalogue.
- *argument: Literal, ...* sont les valeurs qui seront remplacées dans le message.

Exemple

Le fichier KwrCATCatalog.CATNls contient le texte suivant.

```
Zero = "Zero";

Un = "Un /P1";

Deux = "Deux /P1 /P2";
```

Zero, Un et Deux sont les messages. Le premier message ne comporte aucun argument, le deuxième en comporte 1 et le troisième, deux.

Pour afficher ces messages dans une règle Knowledge Advisor, par exemple, écrivez le corps de règle suivant :

```
Message (BuildMessageNLS("KwrCATCatalog","Zero"))
```

OU

```
Message (BuildMessageNLS("KwrCATCatalog","Un",x))
```

Où x est un paramètre.

OU

```
Message (BuildMessageNLS("KwrCATCatalog","Deux",y,z))
```

Où y et z sont des paramètres.



- Notez que si la fonction ne trouve pas la clé ou le catalogue .NLS, elle renvoie une chaîne vide.
- Si le nombre de paramètres est trop élevé par rapport aux arguments du message, les paramètres sont ignorés.
- Si le nombre de paramètres est trop faible par rapport aux arguments du message, les paramètres sont remplacés par la chaîne "???".
- Notez que le fichier .NLS doit être stocké dans la runtime view(dans le répertoire msgcatalog).

Modèle

KwrCATCatalog.CATPart (voir Rule.2)

Direction Constructeurs

Remarque : Les objets de direction ne peuvent pas être créés à l'aide de l'option "Créer paramètre de type" dans *f(x)*, mais vous devez les utiliser dans les autres constructeurs lorsque ces constructeurs doivent être ajoutés comme argument d'un type de direction. Des exemples sont fournis ci-dessous.

Modèles : voir KwrLineConstructors.CATPart and KwrPointConstructors.CATPart

- **direction** (*x*: Length, *y*: Length, *z*: Length): Direction
  Crée une direction définie par l'origine des axes de référence et par le point dont les coordonnées sont définies par l'argument de la fonction.

  Exemple
  ```
  Geometrical Set.1\Point.1   =
  pointtangent(Geometrical Set.1\Spline.1, direction(10mm,10mm,10mm))
  ```

- **direction**(Line) : Direction
  Crée une direction à partir d'une droite.

  Exemple
  ```
  Geometrical Set.1\Point.1   =
  pointtangent(Geometrical Set.1\Spline.1, direction(Geometrical Set.1\line3))
  ```

- **direction**(Plane) : Direction
  Crée une direction à partir d'un plan.

  Exemple
  ```
  Geometrical Set.1\Point.1   =
  pointtangent(Geometrical Set.1\Spline.1, direction(`zx plane`))
  ```

Fonctions Electrical dans Knowledge Products

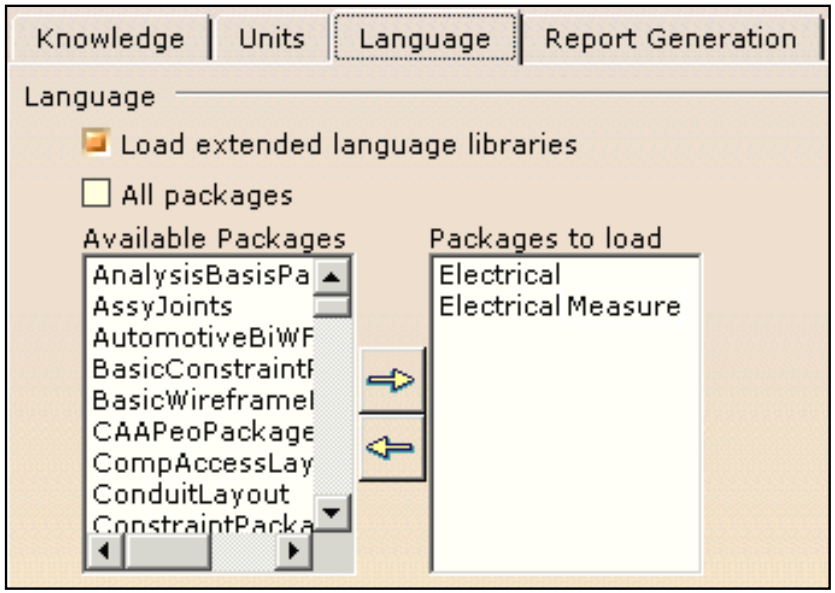
A propos des fonctions Electrical...

- Elec_DistanceCommon
- DistanceWireProduct
- ListAllOuterLayerObjects
- ListDirectlyCoveredObjects
- ListDirectlyOuterLayerObjects
- VisualMode

Pour pouvoir utiliser ces fonctions, vous devez activer le module **ElectricalMeasure**.

Pour ce faire :

1. Sélectionnez **Outils -> Options... -> Général -> Paramètres et mesure** et accédez à l'onglet **Langage**.
2. Choisissez le module **ElectricalMeasure** et cliquez sur la flèche droite :



3. Cliquez sur **OK** pour valider.

Elec_DistanceCommon

Syntaxe

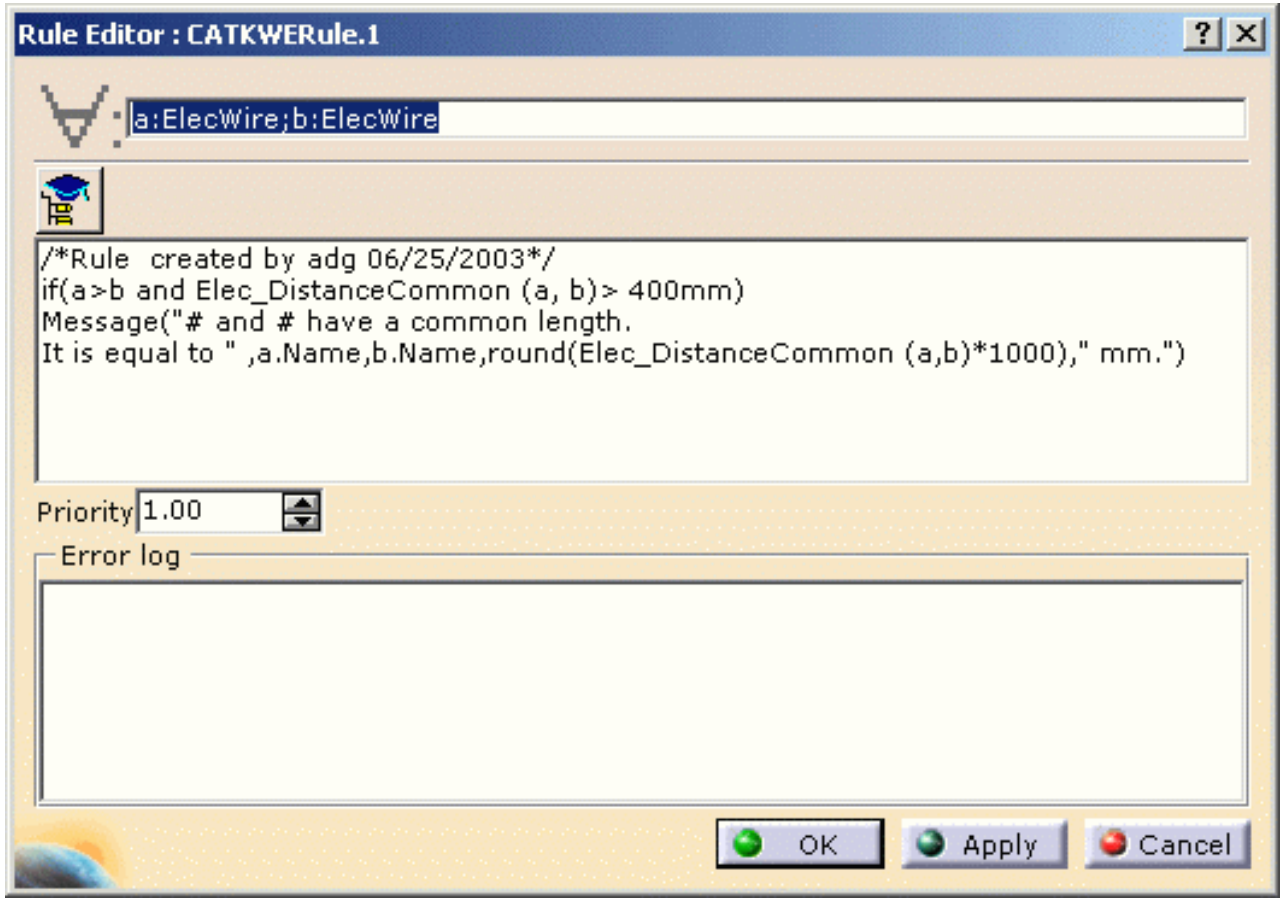
Elec_DistanceCommon(*Wire1 : Fil, Wire2 : Fil*):**Longueur**

Renvoie la longueur commune de deux fils fournis comme arguments d'entrée.

Le type de **Wire1** et **Wire2** est ElecWire.

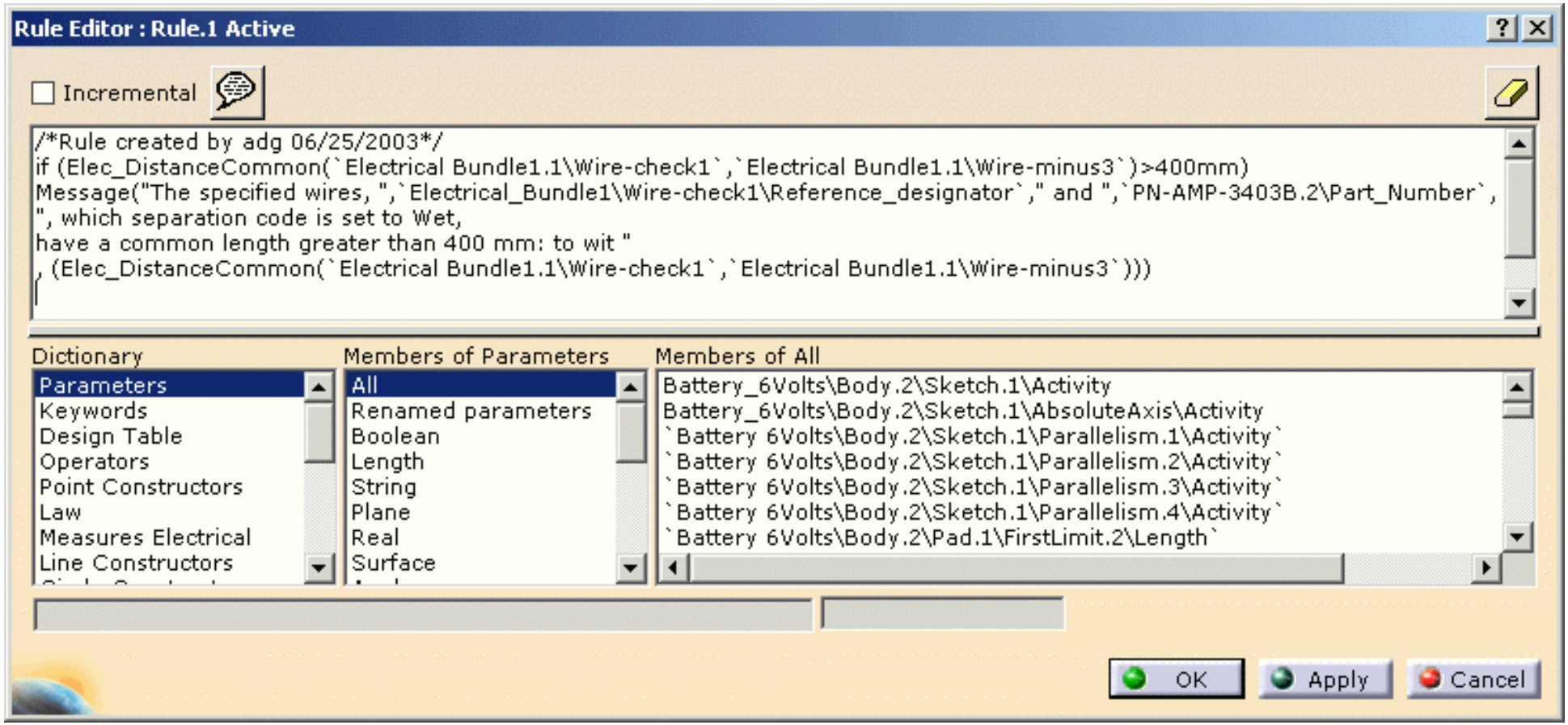
Exemple 1

La fonction utilisateur **Elec_DistanceCommon** peut être utilisée dans Knowledge Expert pour détecter tous les couples de fils de la session présentant une longueur commune supérieure à la valeur donnée.

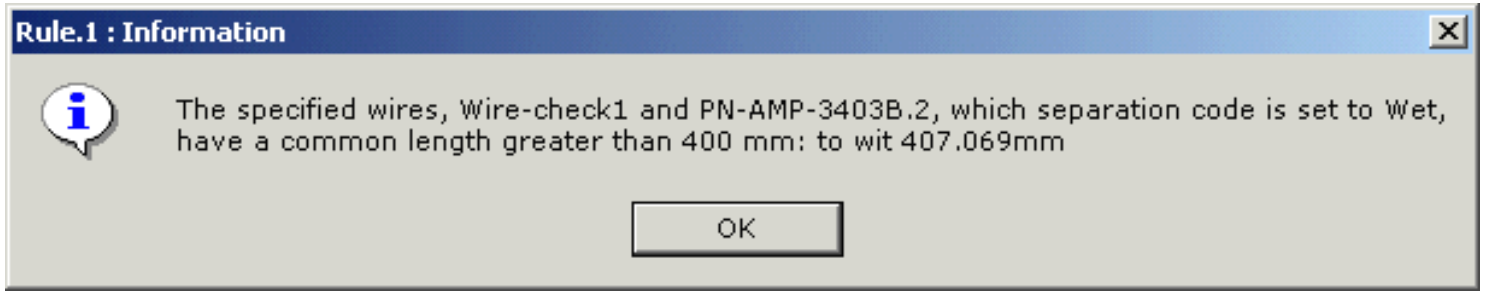


Exemple 2

Dans Knowledge Advisor, elle peut servir à définir une règle indiquant la longueur commune de deux fils spécifiques partageant des propriétés.

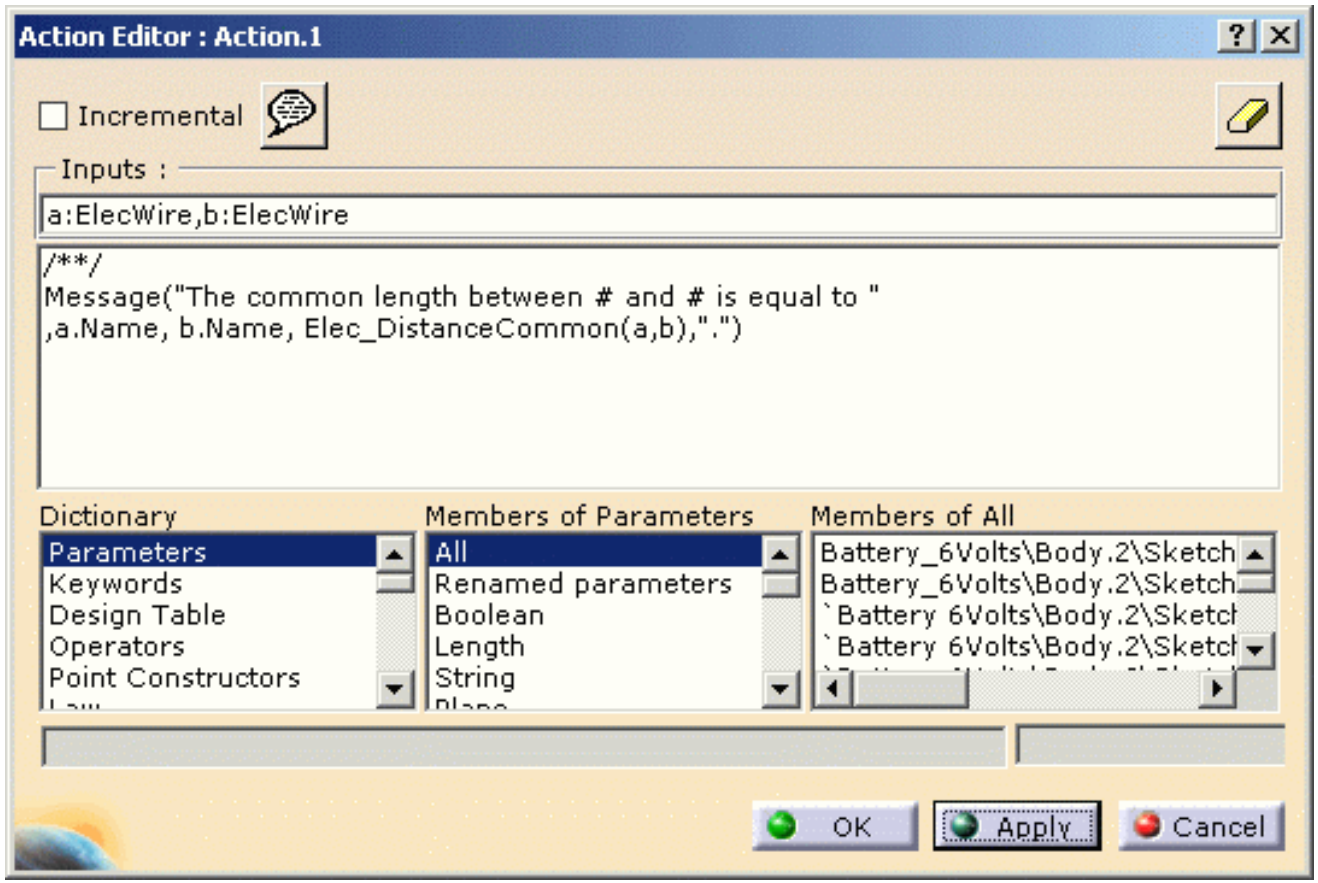


Quand la règle est appliquée, le message suivant apparaît si la condition est satisfaite :

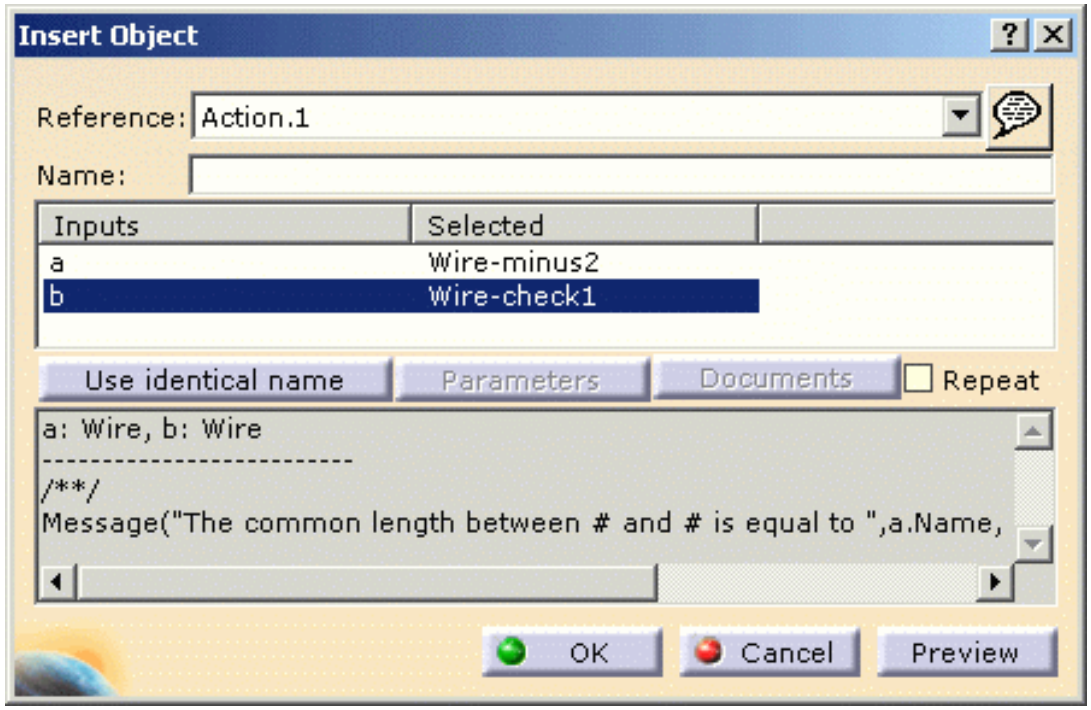


Exemple 3

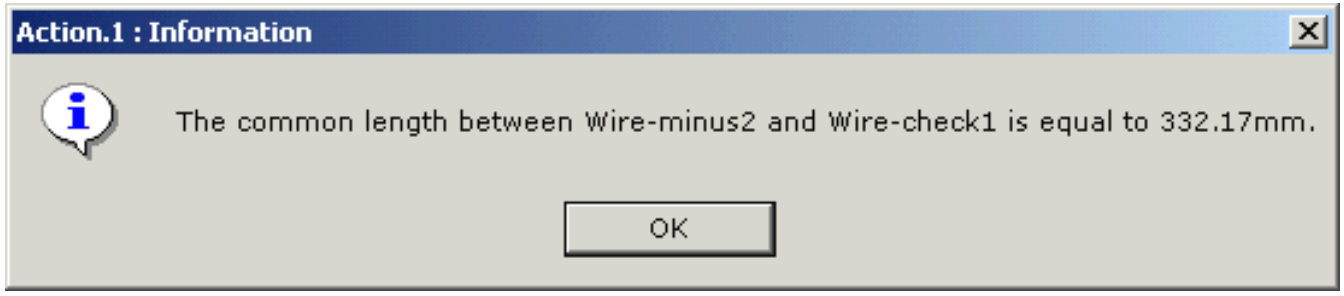
Toujours dans Knowledge Advisor, pour vérifier que deux fils sélectionnés dans l'arbre de spécifications ont une longueur commune, il est possible de définir l'action suivante :



puis de sélectionner deux fils dans l'arbre de spécifications et cliquer sur **OK** pour valider.



Le message suivant s'affiche :



DistanceWireProduct

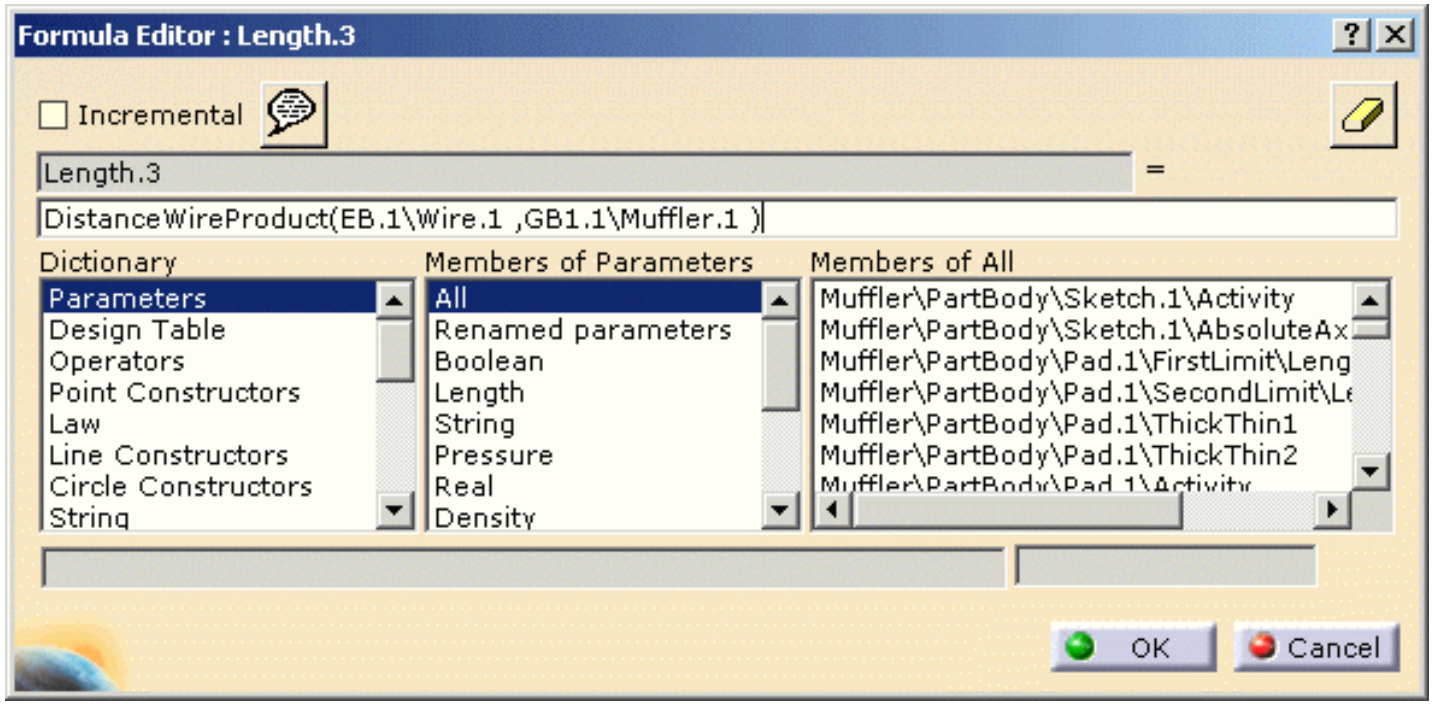
Syntaxe

DistanceWireProduct(*Wire1: Fil, Object: Produit*):**Longueur**

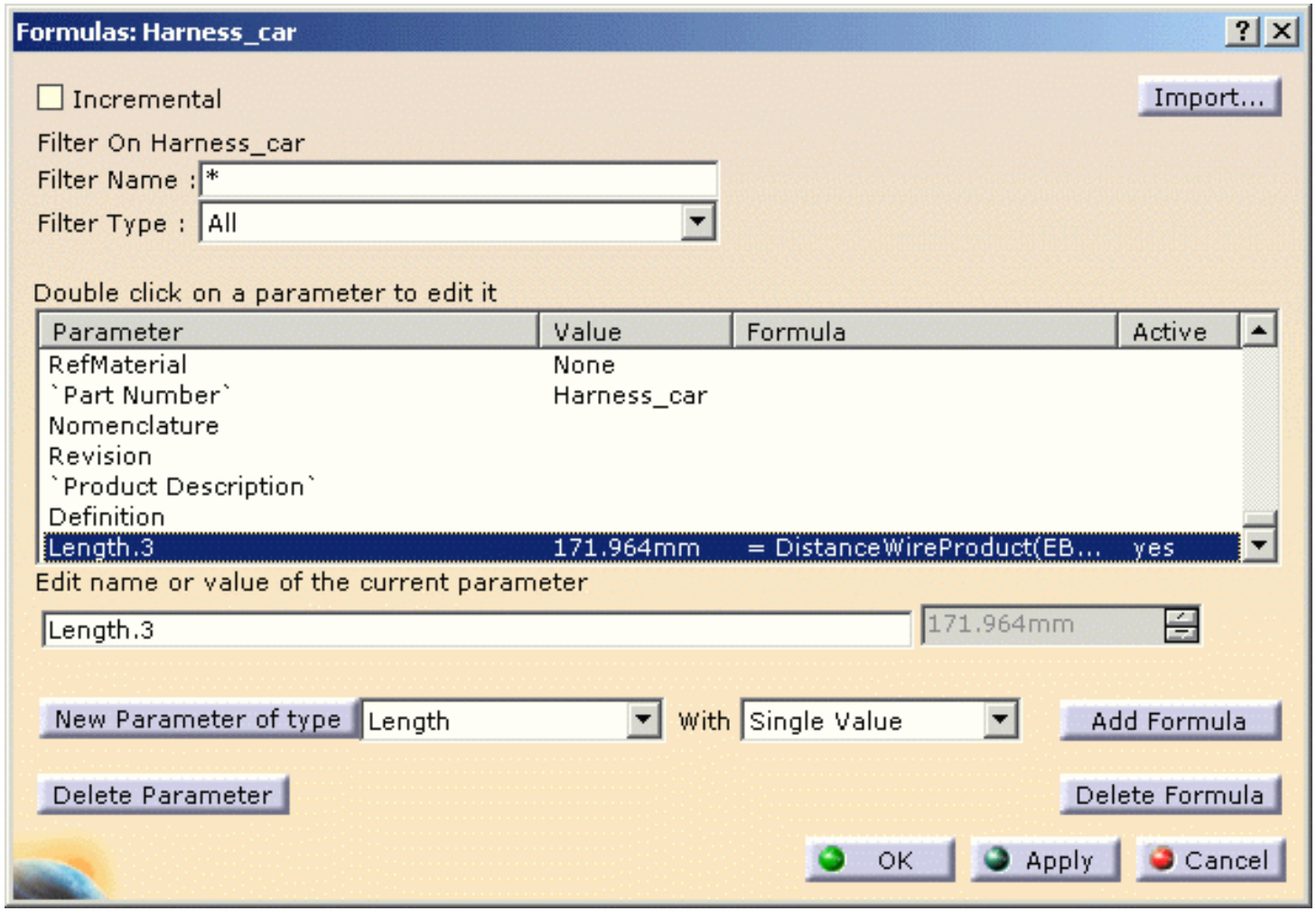
Renvoie la longueur minimale entre un fil et un produit de la session. Le produit doit contenir au moins une pièce.

Exemple 1

Vous pouvez accompagner la fonction utilisateur **DistanceWireProduct** de la commande f(x) pour afficher la distance entre un fil et un composant dans la session.



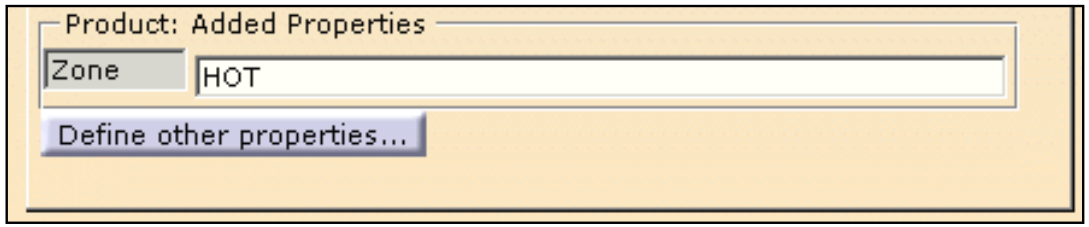
Cette formule renvoie la valeur suivante :



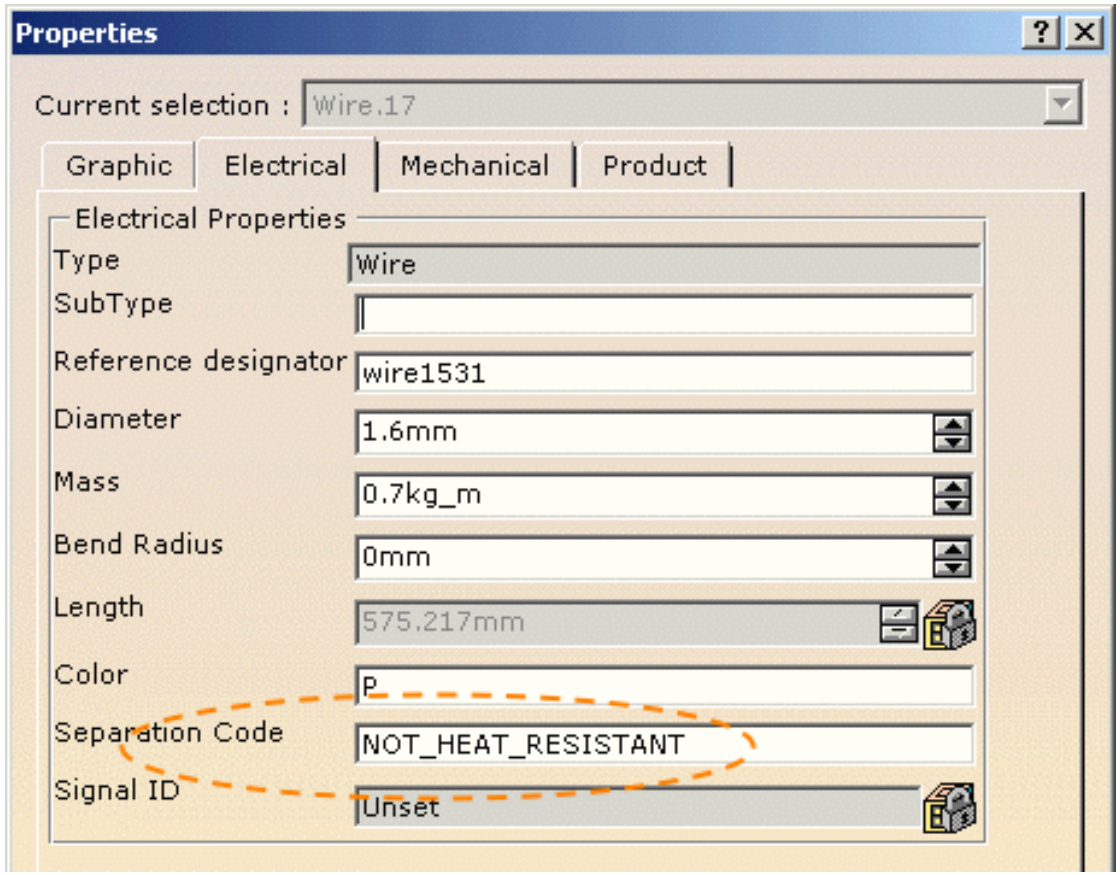
Exemple 2

La fonction utilisateur **DistanceWireProduct** peut être utilisée dans Knowledge Expert pour détecter tous les fils de la session présentant une distance minimale par rapport à des composants définis, qui soit inférieure à une valeur stratégique choisie par l'utilisateur. Les composants peuvent être définis comme thermo-résistants.

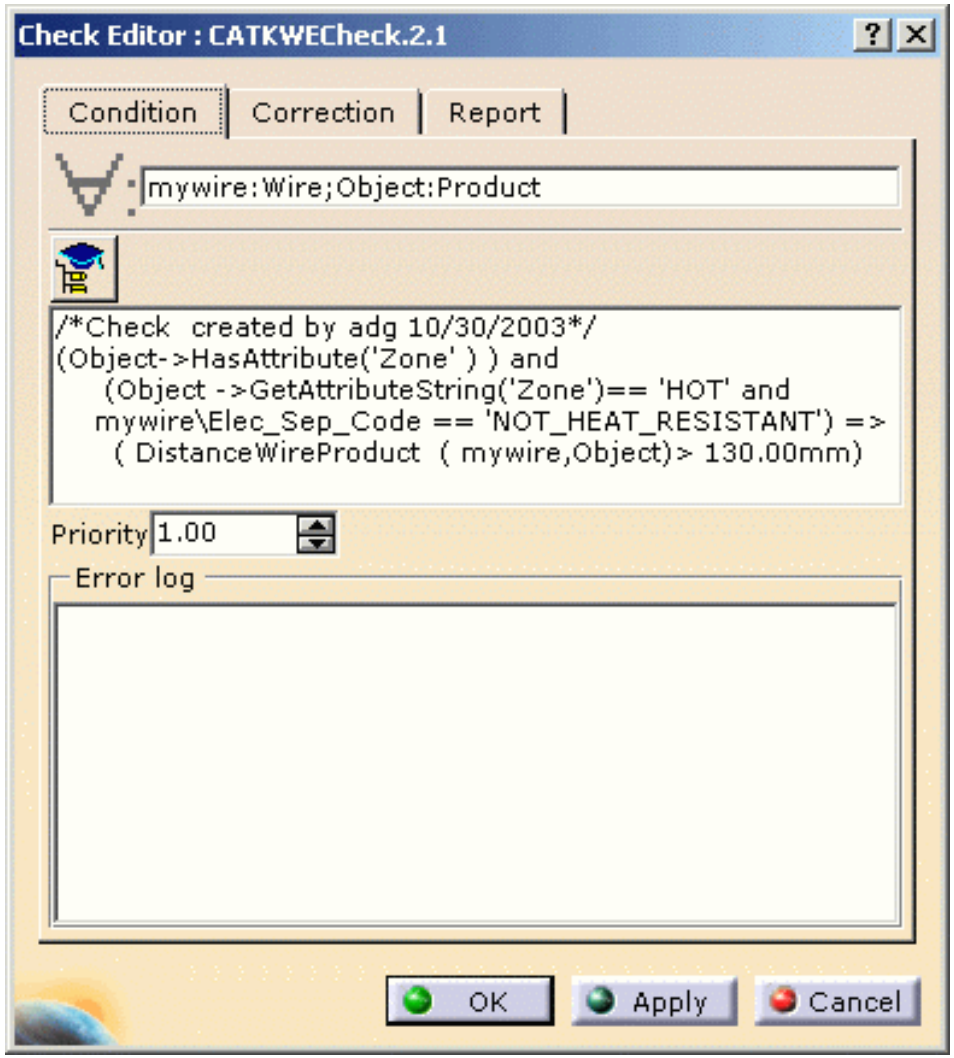
Des propriétés ont été ajoutées au produit :



et aux fils :

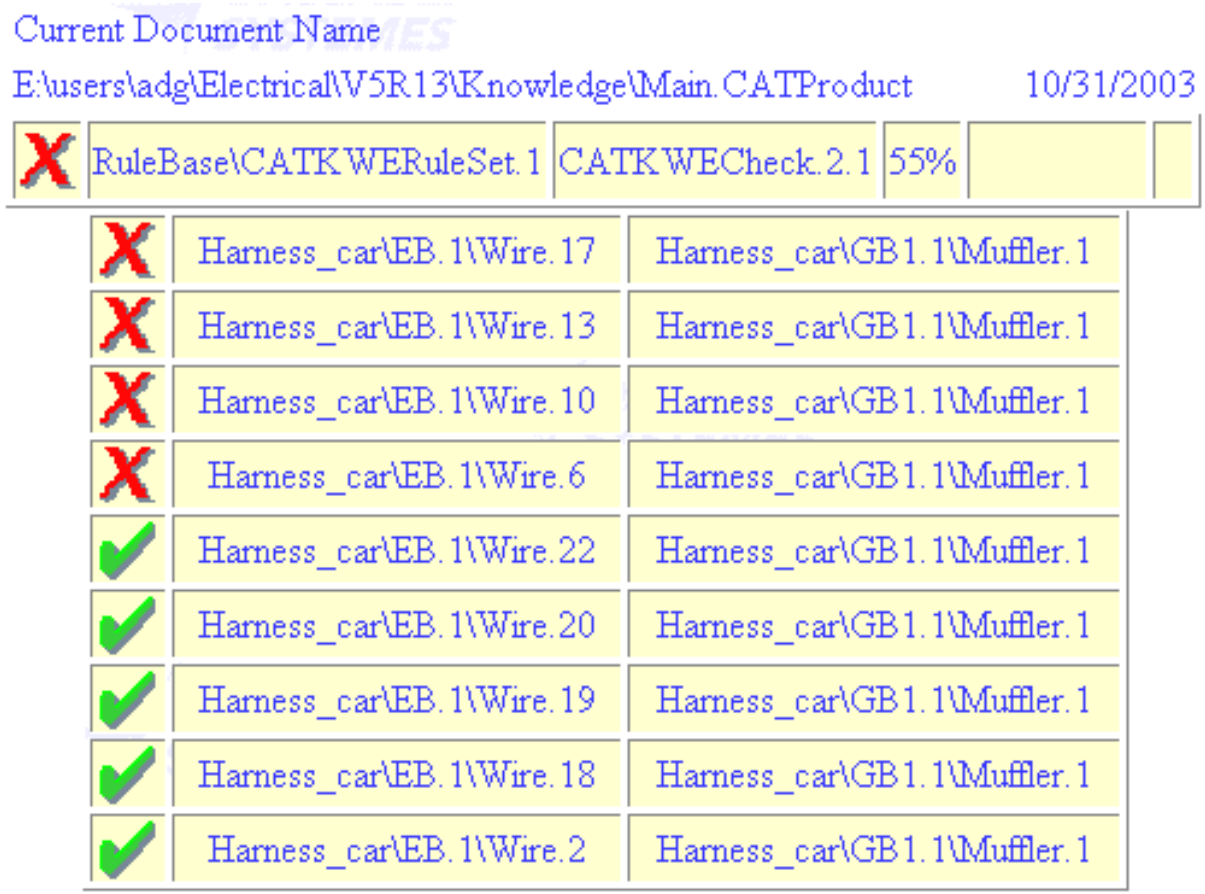


Une vérification est définie comme suit :



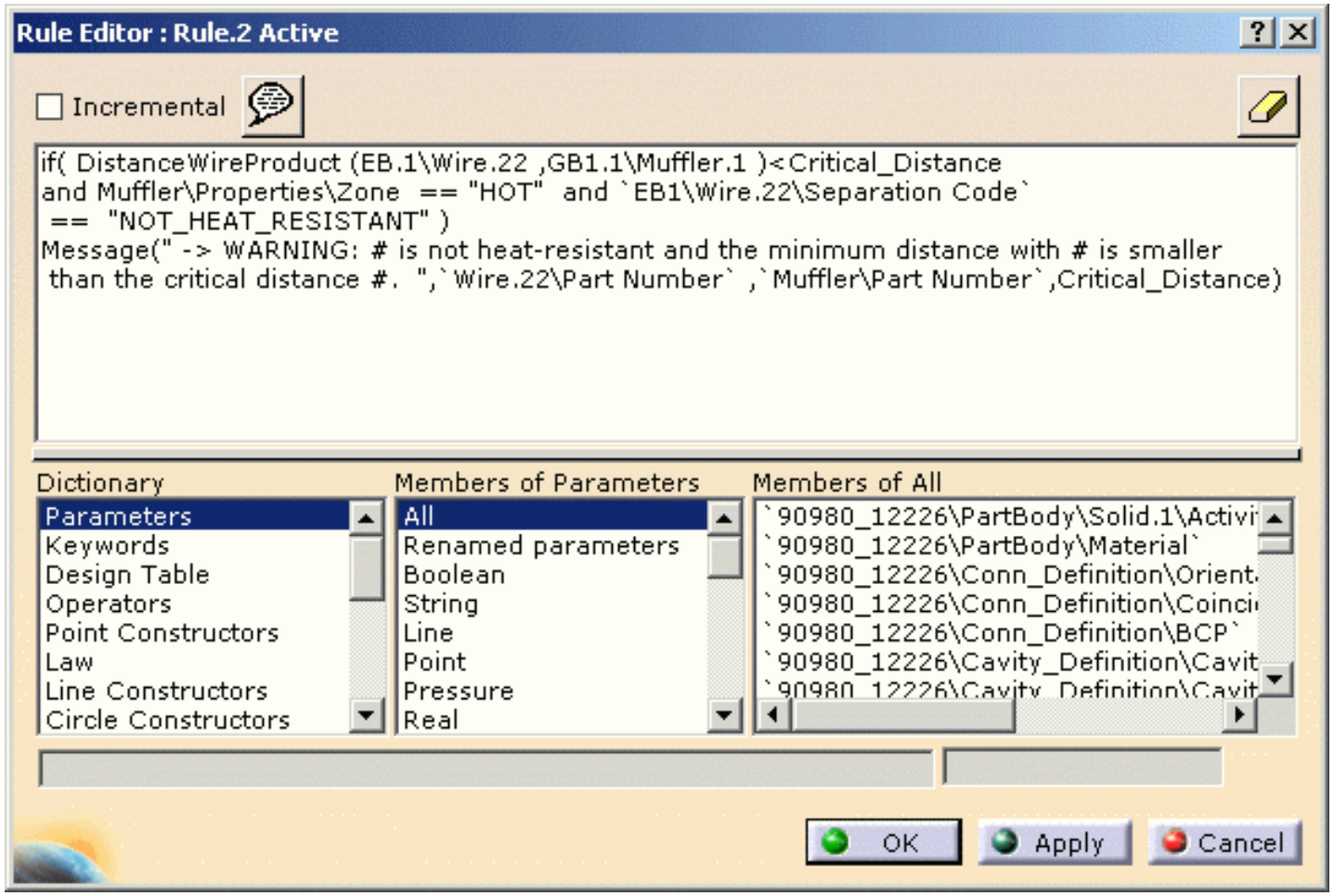
La mise à jour de la session affiche un voyant vert/rouge lors de la vérification : CATKWECheck.2.1

Un rapport affiche le résultat de la vérification : certains fils vérifient la condition et d'autres non.

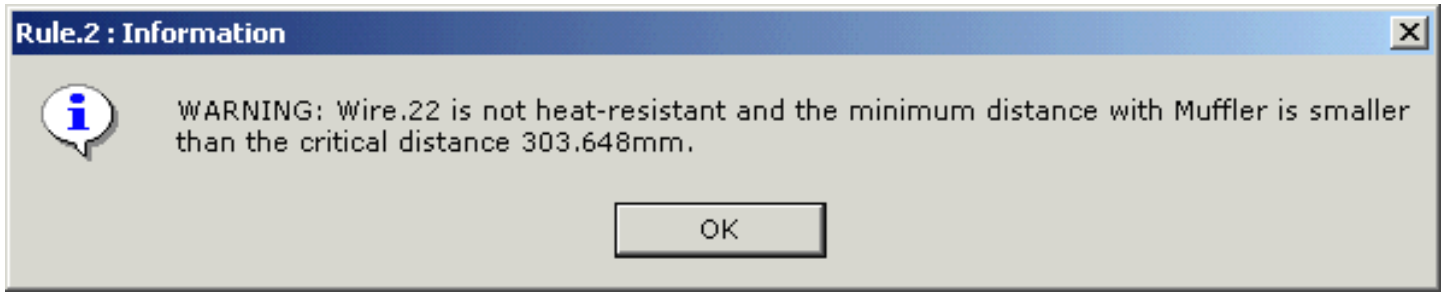


Exemple 3

La fonction utilisateur **DistanceWireProduct** peut être utilisée dans Knowledge Advisor pour définir une règle qui affiche un message d'avertissement si une distance minimale entre un fil et un objet est inférieure à une valeur stratégique choisie par l'utilisateur.

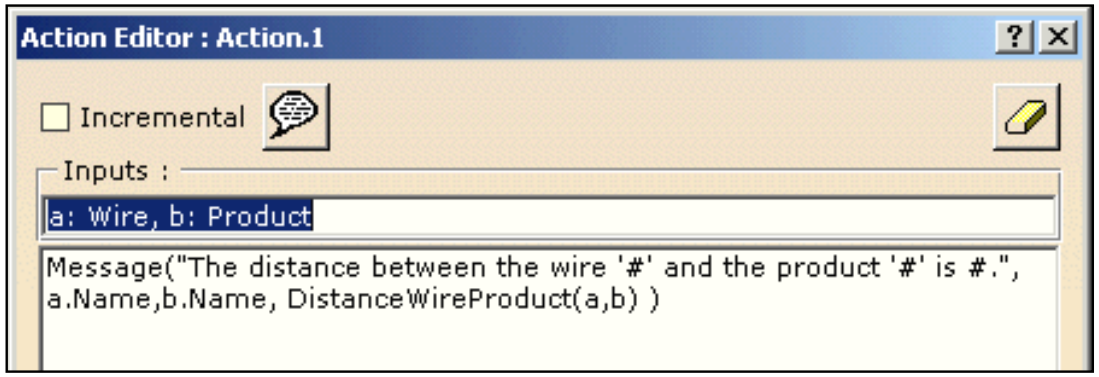


L'exécution de cette règle entraîne l'affichage du message suivant :



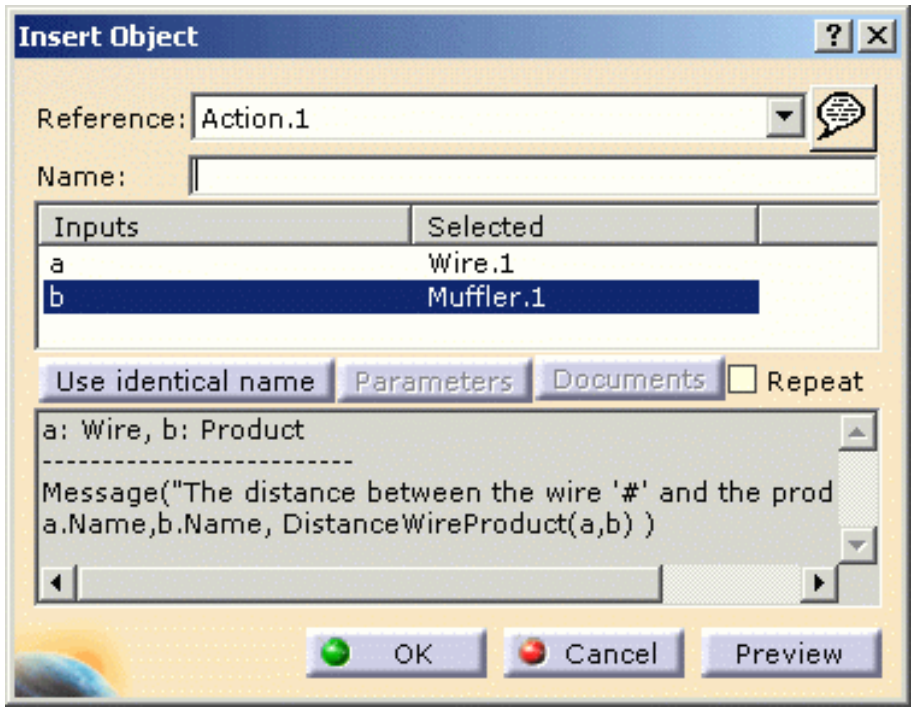
Exemple 4

Toujours dans Knowledge Advisor, il est possible de définir une action à l'aide de la fonction utilisateur **DistanceWireProduct** pour connaître la distance entre un fil et un objet sélectionné dans l'arbre de spécifications :

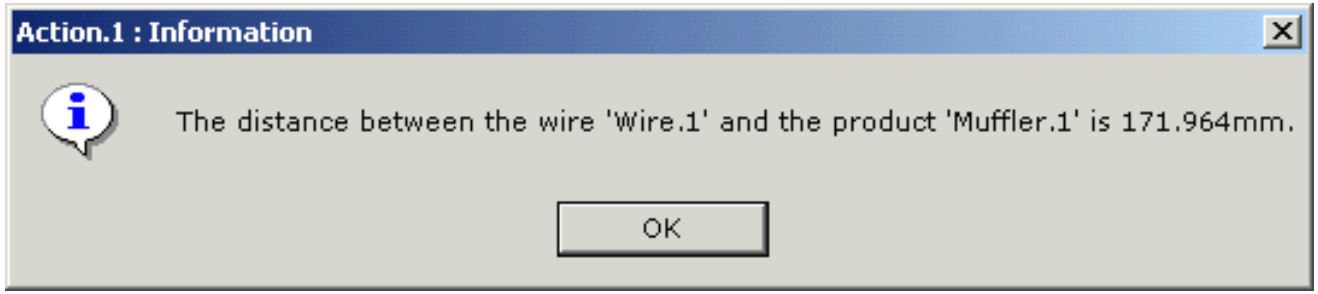


Exécutez l'action à l'aide du menu contextuel**Action.1** :

sélectionnez un fil et un produit dans l'arbre de spécifications, puis validez.



Le message suivant s'affiche :



ListAllOuterLayerObjects

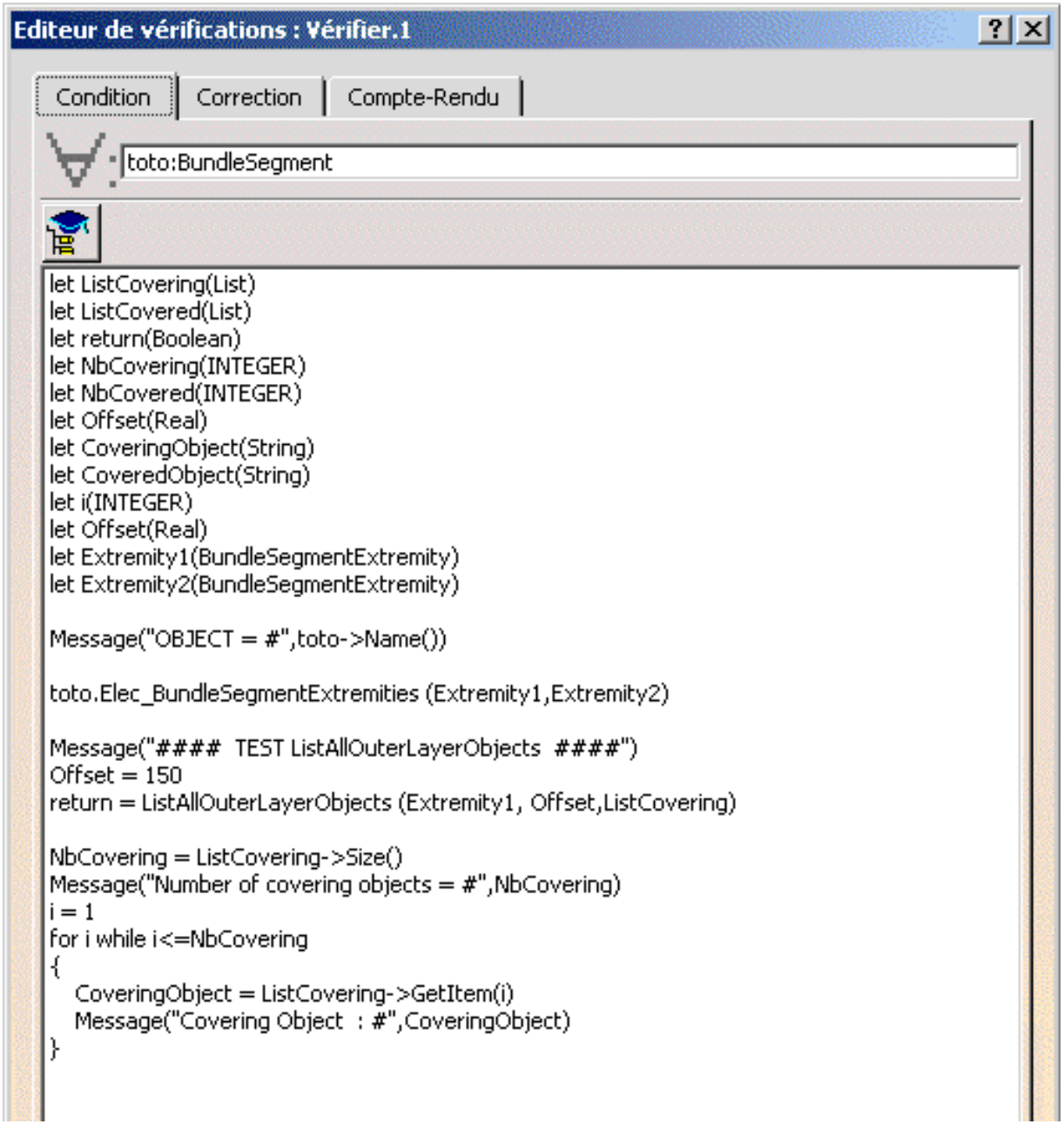
Syntaxe

ListAllOuterLayerObjects (*ElecBundleSegmentExtremity : Composant, Offset : Réel, ListOfAllCoveringProducts(Protections/Supports): out List*) :**Booléen**

Renvoie la liste ordonnée des couvertures de protection et des supports couvrant el segment de harnais au point d'intérêt.
Le point d'intérêt est défini en spécifiant l'extrémité du segment de harnais et en entrant un décalage en mm.

Exemple

Une vérification (à l'aide du langage KWE avancé) pour déterminer les couvertures de protection et les supportes à 150 mm de l'extrémité 1 (Extrémité 1) du segment de harnais s'écrit de la façon suivante :


Editeur de vérifications : Vérifier.1
Condition
Correction
Compte-Rendu
toto:BundleSegment
let ListCovering(List)
let ListCovered(List)
let return(Boolean)
let NbCovering(INTEGER)
let NbCovered(INTEGER)
let Offset(Real)
let CoveringObject(String)
let CoveredObject(String)
let i(INTEGER)
let Offset(Real)
let Extremity1(BundleSegmentExtremity)
let Extremity2(BundleSegmentExtremity)
Message("OBJECT = #",toto->Name())
toto.Elec_BundleSegmentExtremities (Extremity1,Extremity2)
Message("#### TEST ListAllOuterLayerObjects ####")
Offset = 150
return = ListAllOuterLayerObjects (Extremity1, Offset,ListCovering)
NbCovering = ListCovering->Size()
Message("Number of covering objects = #",NbCovering)
i = 1
for i while i<=NbCovering
{
CoveringObject = ListCovering->GetItem(i)
Message("Covering Object : #",CoveringObject)
}

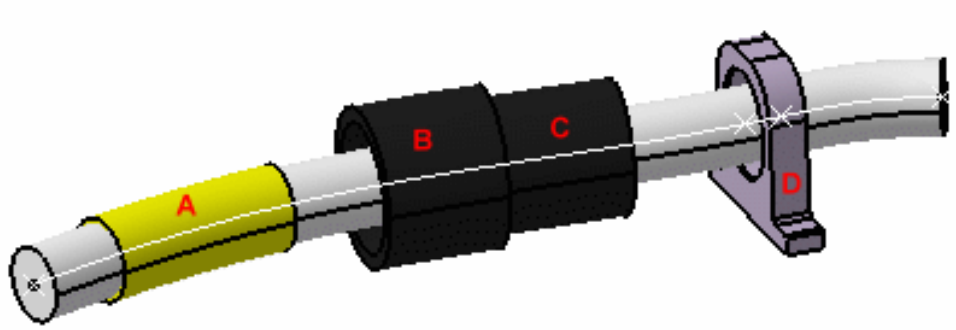
ListDirectlyCoveredObjects

Syntaxe

ListDirectlyCoveredObjects((*Protection/Support) : Composant, ListOfCoveredProducts(Segments de harbnais/Protections/Supports) : out List*) : **Booléen**

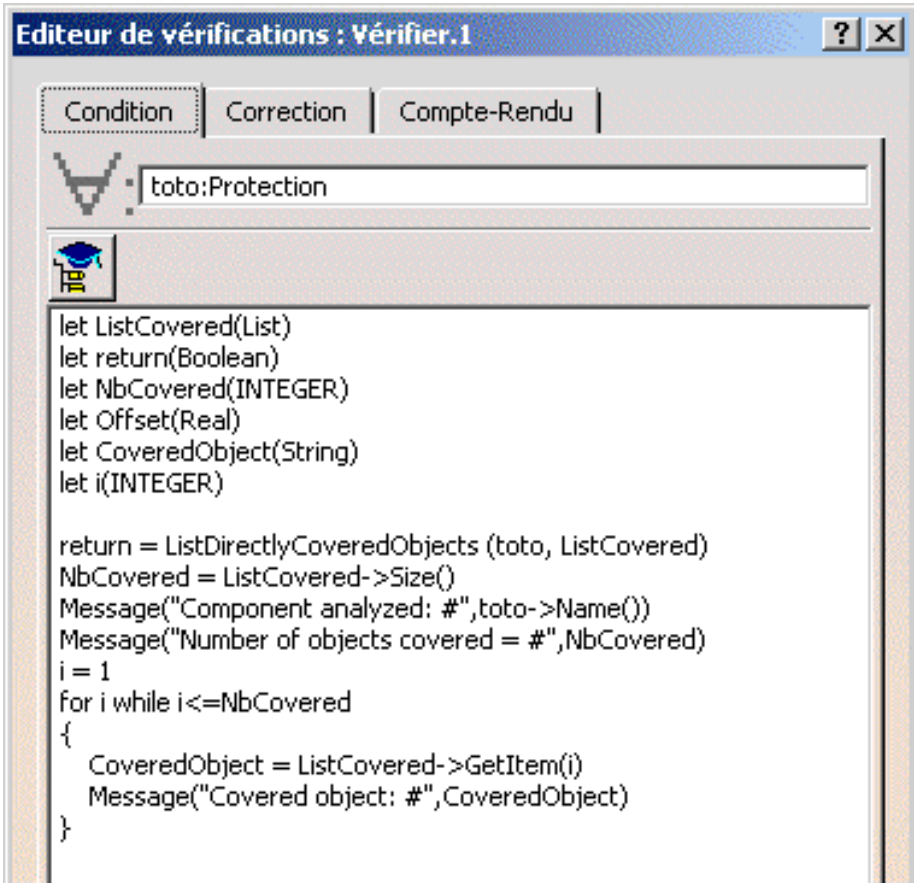
Renvoie la liste des segments de harnais, les couvertures de protection et/ou des supports directement sous le composant (couverture de protection ou support) que vous voulez analyser.
Si le composant que vous voulez analyser n'est pas une couverture de protection ni un support, la liste renvoyée est vide.

Exemple



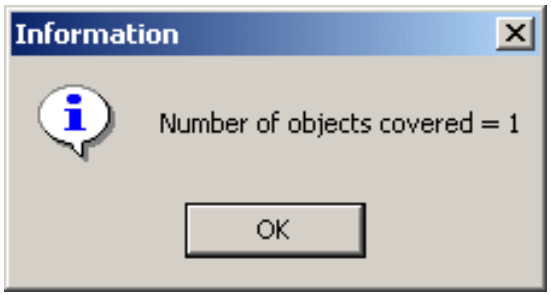
- La bande A est placée directement sur le segment de harnais
- Les tubes ondulés B et C sont liés au segment de harnais et le tube ondulé B recouvre le tube C.
- Le support D standard est lié au segment de harnais.

Une vérification (à l'aide du langage KWE avancé) pour déterminer la liste des segments de harnais, des couvertures de protection et des supports sous le tube ondulé C s'écrit de la façon suivante :



L'exécution de cette vérification renvoie :

- Le nom du composant analysé, dans cet exemple, le tube ondulé C
- Le nombre d'objets recouverts par le tube ondulé C :



- Suivi du nom de ce composant : le segment de harnais.

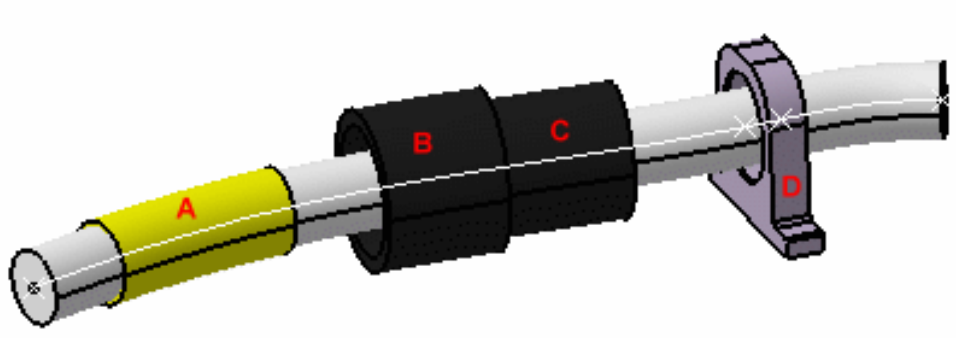
ListDirectlyOuterLayerObjects

Syntaxe

ListDirectlyOuterLayerObjects ((*Segment de harnais/Protection/Support) : Composant, ListOfCoveringProducts(Protections/Supports) : out List*):**Booléen**

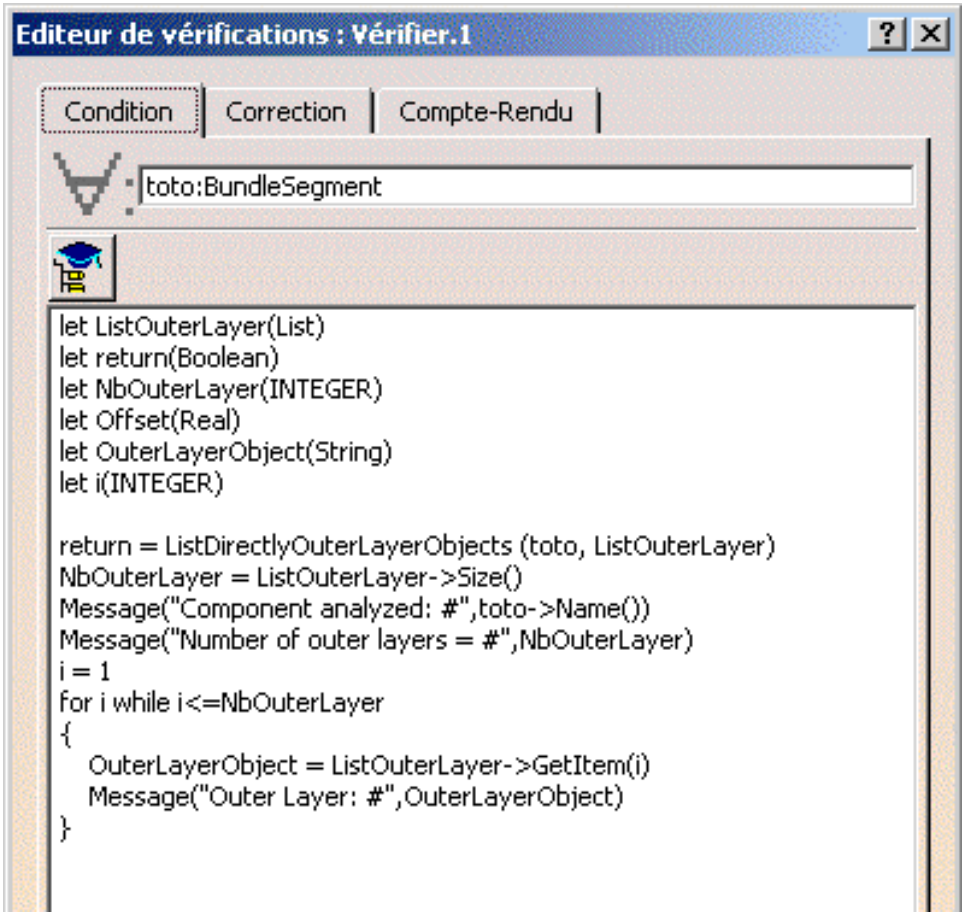
Renvoie la liste des couches extérieures de couvertures de protection et/ou des supports directement sous le composant (segment de harnais, couverture de protection ou support) que vous voulez analyser.
Si le composant que vous voulez analyser n'est ni un segment de harnais, ni une couverture de protection ni un support, la liste renvoyée est vide.

Exemple



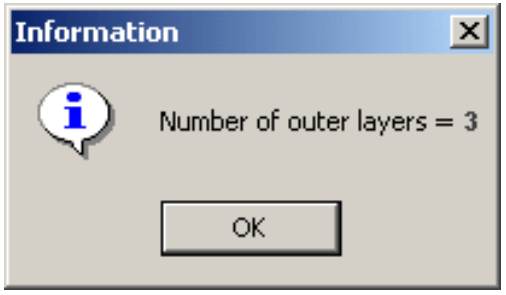
- La bande A est placée directement sur le segment de harnais
- Les tubes ondulés B et C sont liés au segment de harnais et le tube ondulé B recouvre le tube C.
- Le support D standard est lié au segment de harnais.

Une vérification (à l'aide du langage KWE avancé) pour déterminer la liste des couches extérieures qui recouvrent directement le segment de harnais dans l'exemple s'écrit de la façon suivante :



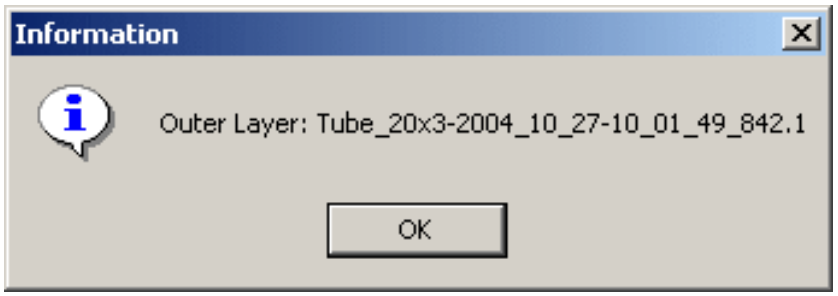
L'exécution de cette vérification renvoie :

- Le nom du composant analysé, dans cet exemple, le segment de harnais.
- Le nombre de couches extérieures sur le segment de harnais :



Trois seulement sont détectées car le tube ondulé B ne recouvre pas directement le segment de harnais puisqu'il recouvre le tube ondulé C.

- Suivi du nom de ces composants :



VisualMode

Syntaxe

VisualMode (*BundleSegment: Composant, Visualization Mode (LIGHT/FULL): Chaîne*): **Booléen**

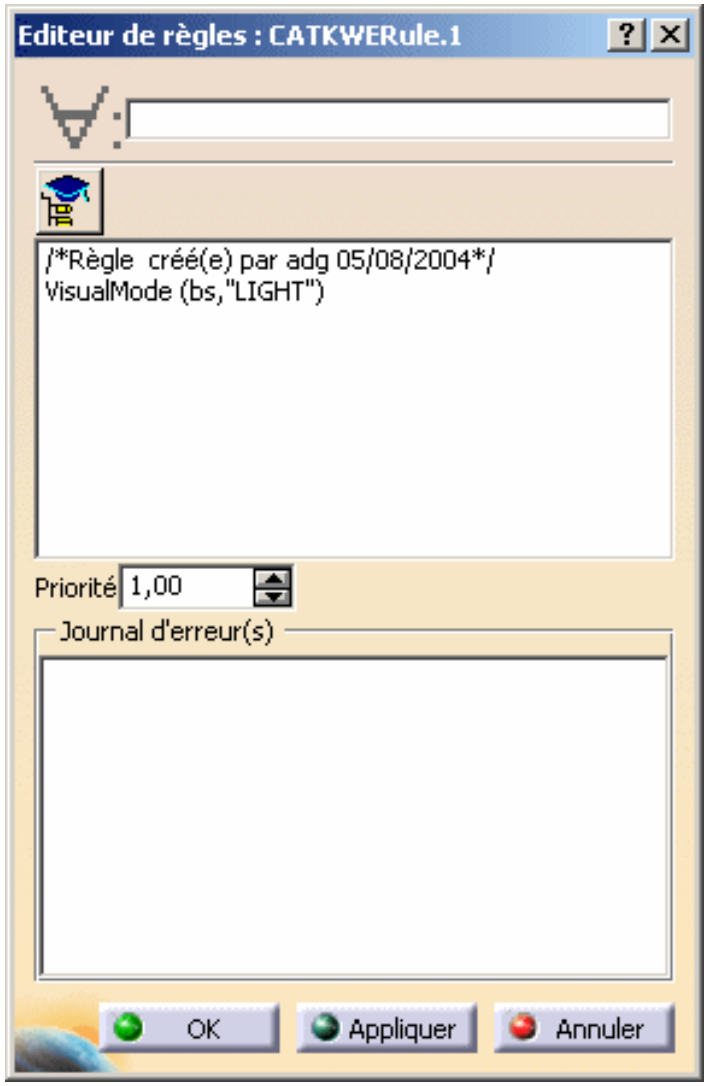
Description

Cette fonction est pratique pour minimiser la taille du harnais en mode **LIGHT**. Elle simplifie la visualisation : seuls la courbe et le diamètre sont représentés, la nervure est supprimée.

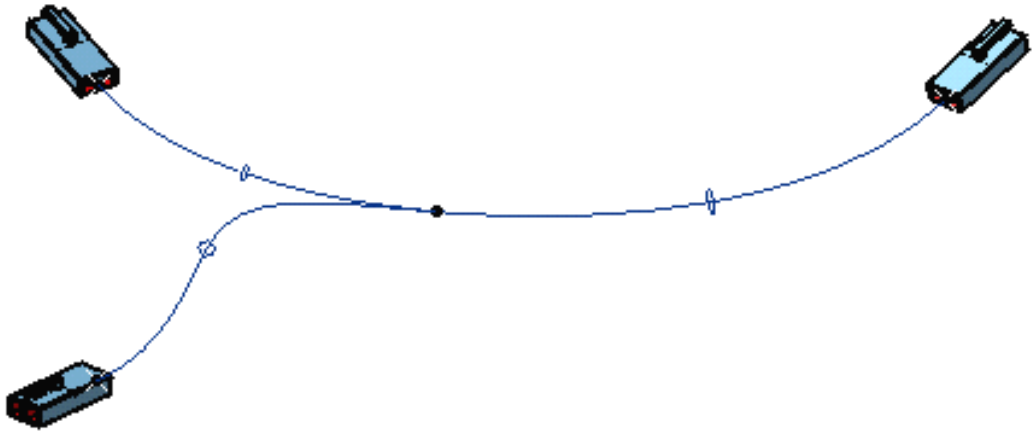
Cette fonction est disponible uniquement pour les segments de harnais appartenant à des documents de segments de harnais branchables.

Exemple

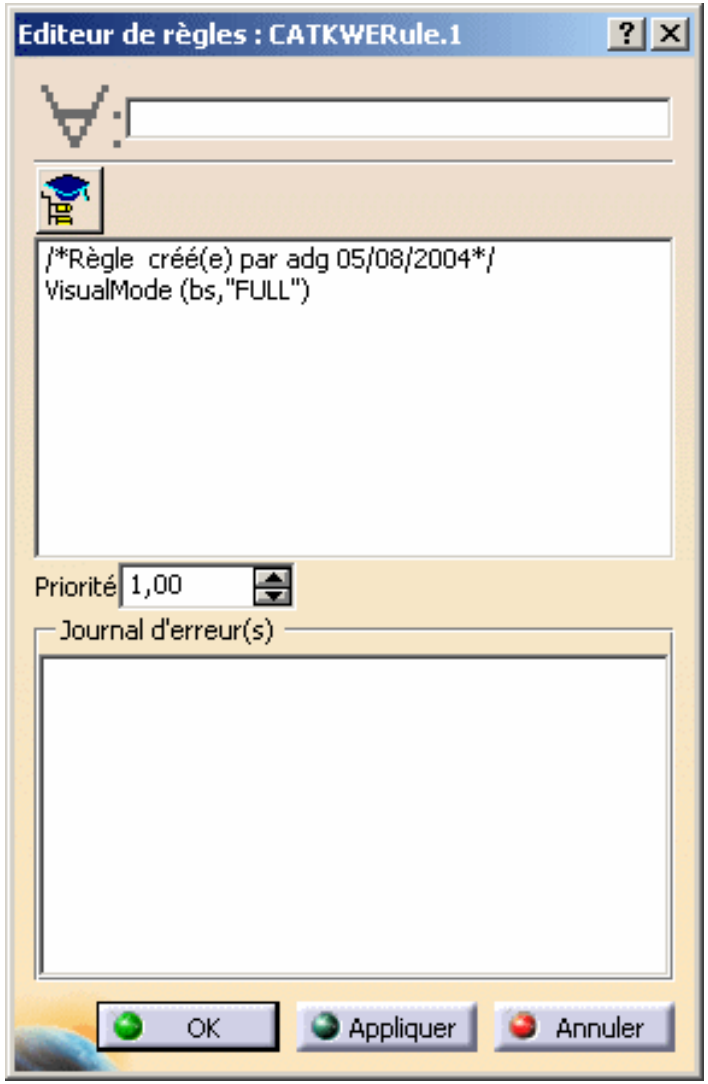
Créez une règle dans Knowledge Expert :



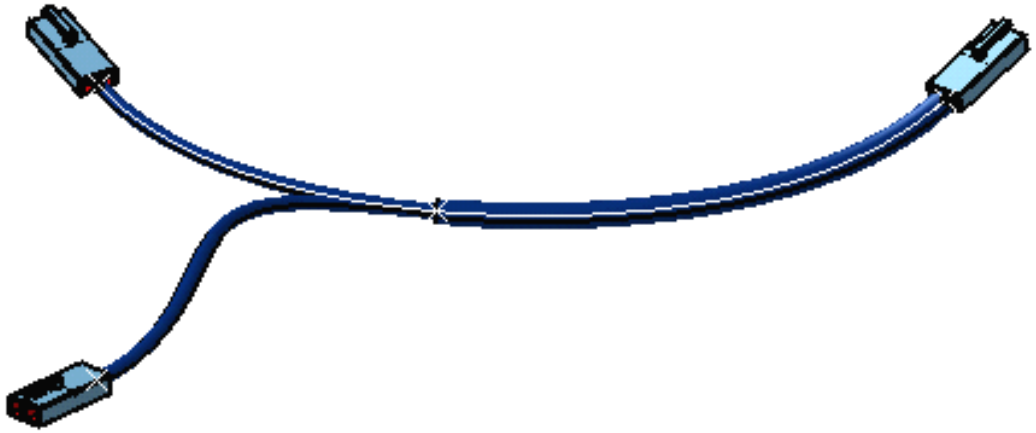
L'exécution de cette règle entraîne l'affichage du harnais en mode **LIGHT** :



La règle est réversible : vous pouvez charger un harnais en mode **LIGHT** et recharger la géométrie en appliquant la règle avec le paramètre **FULL** :



Le harnais s'affiche en mode FULL : la nervure est recréée :



L'utilisation du mode Partiel vous permet de modifier la route et les fils de routage, mais les commandes de coupe, transfert et ajout de points de branche ne sont pas prises en charge.

Pour plus d'informations sur ces commandes, reportez-vous au document Electrical Harness Installation - Guide de l'utilisateur, section Travailler avec des branchements, Scission et Transférer des branchements.

Attribut

Layer	ToleranceMin	ToleranceMax
InferiorRange	SuperiorRange	

Feature.Layer

Méthode permettant d'obtenir/de définir le filtre associé à un composant.



Real.ToleranceMin

Méthode permettant d'accéder à la tolérance minimale pouvant être appliquée aux paramètres de type Longueur et Angle.



Real.ToleranceMax

Méthode permettant d'accéder à la tolérance maximale pouvant être appliquée aux paramètres de type Longueur et Angle.



Real.InferiorRange

Méthode utilisée pour appliquer une fourchette inférieure à un nombre réel.

```
Real.2 =Real.1 .InferiorRange
```



Real.SuperiorRange

Méthode utilisée pour appliquer une fourchette supérieure à un nombre réel.

```
Real.2 =Real.1 .SuperiorRange
```

Utilisation de la fonction de filtrage sur une liste

Les méthodes de listes sont utilisées pour gérer les listes de paramètres, d'extrusion, etc. Elles permettent à l'utilisateur de créer des listes, d'y ajouter des éléments ou d'en supprimer, d'en extraire des valeurs, de déplacer des éléments de la liste vers une autre destination, d'effectuer un filtrage et de copier le contenu d'une liste dans une autre.

- **List.Size** () **:** Integer
  Méthode utilisée pour renvoyer le nombre d'éléments que contient la liste.

- **List.AddItem** (*Object: Objecttype, Index: Integer*):VoidType
  Méthode utilisée pour ajouter un élément à la liste.

```
let list (List)
list.AddItem(PartBody\Trou.2 ,1)
list.AddItem(PartBody\Trou.3 ,2)
Message("#",list.Size())
```

- **List.RemoveItem** (*Index: Integer*) :VoidType
  Méthode utilisée pour supprimer un élément de la liste.

- **List.GetItem** (*Index: Integer*) :ObjectType
  Méthode utilisée pour extraire une valeur/un élément de la liste.

- **List.ReorderItem** (*Current: Integer, Target: Integer*) :ObjectType
  Méthode utilisée pour déplacer un élément de la liste vers une nouvelle destination.

- **Copy** (List: List) : List
  Méthode utilisée pour copier le contenu d'une liste et le coller dans une autre.

- **List** (Next: ObjectType, ...): List
  Méthode utilisée pour créer une liste.

- **List.Sum** (): Real
  Calcule la somme des éléments contenus dans la liste.

- **List.IndexOf** (Element: ObjectType, StartIndex:Integer):Integer
  Renvoie l'index d'un élément de la liste.

- **Compute**()
  Fonction utilisée pour calculer le résultat d'une opération effectuée sur les attributs pris en charge par les entités contenues dans la liste.
  Exemple : List.1 .Compute("+","Hole","x.Diameter",Length.1)
  où :
- List.1 est le nom de la liste concernée par le calcul ;
- + désigne l'opérateur utilisé (les opérateurs pris en charge sont : -, min etmax.).
- "Hole" est le type des éléments de la liste utilisé pour le calcul (vous devez indiquer le type "Hole" pour calculer le diamètre et "Solid" pour calculer le volume)
- x désigne les éléments de la liste. Notez que les éléments contenus dans la liste doivent être de type identique ;
- Length.1 est le paramètre de sortie.



Vous pouvez remarquer que cette méthode n'est disponible que pour les actions.

- **Filter()**
  Méthode utilisée pour filtrer une liste d'objets en extrayant les objets répondant à une expression booléenne.
  Cette méthode présente la signature suivante :
  List.Filter(type de chaîne,expression de chaîne) : Liste
  - Type correspond au type d'objets que l'utilisateur veut extraire (il peut s'agir de "". Dans ce cas, aucun filtrage n'est effectué sur les types)
  - L'expression de chaîne correspond à l'expression booléenne à laquelle doivent répondre les objets de ce type donné. Dans cette expression "x" est utilisé comme nom de variable de Type. Cette chaîne peut être égale à "". Dans ce cas, aucune expression n'est vérifiée.

Exemple :
```
I = (List.Filter("Hole","x.Diameter > 3mm")).Size()
I = (List.Filter("Hole","")).Size()
```



Vous pouvez remarquer que cette méthode n'est disponible que pour les actions.

Utilisation de la fonction de périmètre

Les mesures sont des fonctions permettant de calculer un résultat à partir de données capturées dans la zone géométrique. Ces mesures sont des objets liés aux applications et n'apparaîtront pas dans le dictionnaire si le programme installé n'est pas correct (par exemple, Part Design ou Generative Shape Design).

Modèle : KwrMeasuresWiz.CATPart

- **distance***(Body1, Body2)* **:** Length
  Renvoie la distance entre deux corps d'une pièce.

- **minimumCurvatureRadius***(Curve):Length*
  Pour un élément de dimension 1 (une courbe), permet à l'utilisateur de mesurer son rayon de courbure minimum.

- **nbDomains***(Body)***: Integer**
  Pour tous les types d'élément, permet à l'utilisateur de calculer le nombre de domaines.

- **length**(*GSMCurve*) :**Length**
  Renvoie la longueur totale d'une courbe.

- **length**(*GSMCurve*, *Point1*, *Point2*) : **Length**
  Renvoie la longueur du segment de courbe situé entre *Point1* et *Point2*.

- **length**(*GSMCurve*,*Point1*,*Boolean*): **Length**
  Renvoie la longueur du segment de courbe situé entre *Point1* et l'une des extrémités de la courbe.
  Si vous modifiez la valeur booléenne, vous pouvez extraire la longueur entre le point défini et l'autre extrémité.

- **area**(*Surface*): Area
  Renvoie l'aire d'une surface créée par le programme Generative Shape Design (par exemple, une surface extrudée).

- **area**(*Curve*) : Area
  Renvoie l'aire délimitée par une courbe.

- **perimeter**(*Surface*,...):Length
  Renvoie le périmètre d'une surface. Plusieurs entités de surface peuvent être indiquées en entrée. La fonction de périmètre résume le périmètre de chaque surface. La valeur renvoyée est une longueur.

- ***point*.coord**(*Integer*): Length
  Renvoie les coordonnées d'un point. Renvoie X si la valeur indiquée est 1, Y si la valeur indiquée est 2 et Z si la valeur est 3.

- **Body.centerofgravity**(x: out length, x: out length, z: out length): Void Type
  Permet à l'utilisateur de calculer le centre de gravité.

- **volume**(*Volume geometry,...*) : Volume
  Renvoie le volume d'un volume.

- **volume**(*closedSurface*) : Volume
  Renvoie le volume d'une surface fermée.

- **angle**(*Center: Point1, Pt1: Point2, Pt2*) : Angle
  Renvoie l'angle mesuré entre les droites "C-Point1" et "C-Point2".

- **angle**(*Direction,Direction*) : Angle
  Renvoie l'angle entre deux directions.

- **angle**(*Line, Line*) : Angle
  Renvoie l'angle mesuré entre les droites *Line1* et *Line2*.

- **angle**(*Plane, Plane*) : Angle
  Renvoie l'angle entre les 2 plans.

- **angleoriented***(Direction, Direction, Direction)***: Angle**
  Renvoie l'angle entre 2 directions et orienté par une troisième direction.

- **angleoriented**(*Line, Line, Direction*)**: Angle**
  Renvoie un angle entre 2 droites et orienté par la direction.

- **angleoriented**(*Plane, Plane, Direction*)**: Angle**
  Renvoie un angle entre 2 plans et orienté par la direction.

- **curvature**(*crv*: Curve, *pt*: Point): Real
  Renvoie la courbure d'une courbe en un point donné.

- **distancedir** (*Body, Body, Direction*): Length
  Renvoie la distance entre deux corps d'une pièce et orientée par la direction.

Surface Constructeurs

Offset	assemble	split (surface, surface, boolean)
split (surface, curve, boolean)	trim(*surface*, *boolean*, *surface*, *boolean*)	near(surface, wireframe) : Surface
extrude(*curve*, *direction*, *length*, *length*, *boolean*) : Surface	extrude(*surface*, *direction*, *length*, *length*, *boolean*) : Surface	revolve(*curve*, *line*, *angle*, *angle*) : Surface
revolve(*surface*, *line*, *angle*, *angle*) : Surface	loft(*sections*: list, *orientations*: list)	loft(*sections:* list, *orientations*: list, *guides*: list)

Modèle : KwrSurfaceConstructors

- **offset**(*surface*, *length*, *boolean*) : Surface
  Crée une surface décalée. Pour créer la surface de l'autre côté de la surface de référence, affectez la valeur false à l'argument booléen orientation.

- **assemble**(*surface*, ...) : Surface
  Crée une liaison entre plusieurs surfaces.

- **split**(*surface*, *surface*, *boolean*) : Surface
  Crée une division d'une surface par une autre. Utilisez le troisième argument pour sélectionner le côté à conserver.

- **split**(*surface*, *curve*, *boolean*) : Surface
  Crée une division d'une surface par une courbe. Utilisez le troisième argument pour sélectionner le côté à conserver.

- **trim**(*surface*, *boolean*, *surface*, *boolean*) : Surface
  Crée une découpe d'une surface par une autre. Utilisez les arguments booléens pour sélectionner le côté à conserver sur chaque surface.

- **near**(surface, wireframe) : Surface
  Extrait un sous-élément connexe de l'entité non connexe la plus proche d'un autre élément.

- **extrude**(*curve*, *direction*, *length*, *length*, *boolean*) : Surface
  Crée un profil de représentation filaire (wireframe) par extrusion dans une direction donnée.

- **extrude**(*surface*, *direction*, *length*, *length*, *boolean*) : Surface
  Crée une surface dans une direction donnée. Le résultat est la peau du volume généré.

- **revolve**(*curve*, *line*, *angle*, *angle*) : Surface
  Fait pivoter un profil wireframe autour d'un axe donné.

- **revolve**(*surface*, *line*, *angle*, *angle*) : Surface
  Fait pivoter une surface autour d'un axe donné. Le résultat est la peau du volume généré.

- **loft**(*sections*: list, *orientations*: list)
  Crée une surface guidée à partir de plusieurs sections.

- **loft**(*sections:* list, *orientations*: list, *guides*: list)
  Crée une surface guidée à partir de plusieurs sections et de plusieurs guides.

Wireframe Constructeurs

spline	intersect	intersect	intersect
curveparallel	project	project	project
assemble	corner	split	trim
near	near	extrude	revolve

Modèle : KwrWireFrameConstructors.CATPart

- **spline**(pt: Point, ...): Curve
  Crée une spline à partir de plusieurs points.

- **intersect**(*crv*: Curve, *crv*: Curve) : Point
  Crée le point d'intersection entre deux courbes.

- **intersect**(*crv*: Curve, *su*: Surface) : Point
  Crée le point d'intersection entre une courbe et une surface.

- **intersect**(*su*: Surface, *su*: Surface) : Curve
  Crée la courbe d'intersection entre deux surfaces.

- **curveparallel**(*crv*: Curve, *su*: Surface, offset: Length) : Curve
  Crée la courbe parallèle à une autre. La surface définie par le deuxième argument est le support.

- **project**(toproject: Point, support: Curve): Point
  Projette un point sur une courbe.

- **project**(toproject: Point, support: Surface): Point
  Projette un point sur une surface.

- **project**(toproject: Point, support: Surface): Surface
  Projette une courbe sur une surface.

- **assemble**(Curve,...):Curve
  Crée une liaison entre plusieurs courbes.

- **corner**(*crv1*: Curve, *crv2*: Curve, *support*: Surface, radius: Length, orientationcrv1: Boolean, orientationcrv2: Boolean, trim: Boolean : Curve
  Crée un coin entre deux courbes. Les arguments 5 et 6 doivent être utilisés pour analyser les solutions possibles. Pour plus d'informations sur les coins, reportez-vous au document *Generative Shape Design - Guide de l'utilisateur*.

- **split**(tosplit:curve, splitting: Wireframe, orientation:Boolean):Curve
  Permet de découper une surface.

- **trim**(crv1: Curve, orientationCrv1: Boolean, orientationCrv1: Boolean, crv1: Curve, orientationCrv2, Boolean): Curve
  Permet de relimiter deux éléments filaires.

- **near**(crv:Curve, near:Wireframe): Curve
  Crée l'entité la plus proche de plusieurs sous-éléments. Le résultat est une courbe.

- **near**(crv:Point, near:Wireframe): Point
  Crée l'entité la plus proche de plusieurs sous-éléments. Le résultat est un point.

- **extrude**(Point, Direction, length1: Length, length2: Length, orientation: Boolean): Line
  Crée une droite. Extrusion d'un point dépendant d'une direction.

- **revolve**(Point, axis, Line: angle1, Angle: angle2, Angle): Circle
  Permet de créer un cercle en faisant tourner un point selon une direction donnée.

Mesures de pièces

Les éléments smartVolume et smartWetarea font référence aux états intermédiaires d'un solide. L'élément smartVolume ne calcule pas le volume de chaque extrusion contenue dans une entité PartBody, mais le volume total de cette dernière.
Exemple : Pour une entité PartBody contenant 3 extrusions : le volume de Pad.1 = 0,1 m3, le volume de Pad.2 = 0,1 m3 et le volume de Pad.3 = 0,1 m3. Le volume de Pad.3 affiché sera Pad.3=0,3 m3. Le volume de Pad.3 = le volume de Pad.1 + le volume de Pad.2 + le volume de Pad.3.

Remarquez que cette égalité s'applique également à l'élément smartWetarea (la surface mouillée est calculée dans sa totalité).

- **smartVolume***(elem: Solid, ...)*: Volume
  Renvoie le volume d'un solide.

  Exemple
  Total_Volume=
  smartVolume(PartBody)

- **smartWetarea***(elem: Solid, ...)* : Area
  Renvoie la surface mouillée d'un solide.

  Exemple
  Total_Area=
  smartWetarea(PartBody\Pad.1)

Plans Constructeurs

Modèle : KwrPlaneConstructors.CATPart

- **plane**(*point*, *point*, *point*) : Plane
  Crée un plan passant par trois points.
- **plane**(*a*:Real, *b*:Real, *c*:Real, *d*:Length) : Plane
  Crée un plan à partir de son équation $aX+bY+cZ=d$.
- **plane**(*line*, *line*) : Plane
  Crée un plan passant par deux droites.
- **plane**(*point*, *line*) : Plane
  Crée un plan passant par un point et une droite.
- **plane**(*curve*) : Plane
  Crée un plan contenant une courbe plane.
- **planetangent**(*surface*, *point*) : Plane
  Crée un plan tangent à une surface en un point.
- **planenormal**(*curve*, *point*) : Plane
  Crée un plan normal à une courbe en un point.
- **planeoffset**(*plane*, *length*, *boolean*) : Plane
  Crée un plan décalé d'une distance donnée par rapport à un autre. Pour créer le plan de l'autre côté du plan de référence, affectez la valeur false à l'argument booléen orientation.
- **planeoffset**(*plane*, *point*) : Plane
  Crée un plan décalé d'un point donné par rapport à un autre.
- **planeangle**(*plane*, *line*, *angle*, *boolean*) : Plane
  Crée un plan angulaire. Pour créer le plan de l'autre côté du plan de référence, affectez la valeur false à l'argument booléen orientation.
- **planemean**(Point,...): Point
  Calcule un plan moyen à partir d'un ensemble de points.

Opérateurs d'analyse

- **energy** (*Case: StaticSolution*)
  Calcule l'énergie globale dans une solution statique.

- **misesmax** (*Case: StaticSolution*)
  Calcule la valeur maximale du la contrainte VonMises en noeuds.
  Exemple
  misesmax.1=misesmax("Finite Element Model\Static Case Solution.1")

- **dispmax** (*Case: StaticSolution*)
  Calcule le déplacement maximum en noeuds.
  Exemple
  length.1=dispmax("Finite Element Model\Static Case Solution.1")

- **frequency** (*Case: FrequencySolution*)
  Calcule une fréquence donnée.
  Exemple
  Frequency.1=Frequency("Finite Element Model\Frequency Case Solution.1")

- **frequencies** (*Case: FrequenciesSolution*)
  Calcule toutes les fréquences.
  Exemple
  FrequenciesList.1=Frequencies("Finite Element Model\Frequencies Case Solution.1")

- **globalerror** (*Case: StaticSolution*)
  Calcule le pourcentage d'erreur global pour un cas statique.
  Exemple
  percentage.1=globalerror("Finite Element Model\Static Case Solution.1")

- **bucklingfactors** (*Case: BucklingSolution*)
  Calcule la liste des facteurs de flambement (déformation).
  Exemple
  Bucklingfactors.1=BucklingFactors("Finite Element Model\Buckling Case Solution.1")

- **dispmaxongroup** (*Case: AnalysisResults, Group:Group*): Length
  Calcule le déplacement maximum en noeuds. S'applique à un groupe d'éléments.

- **reaction** (*Entity: EntityForReaction, Case: StaticSolution, Axis:Axis System)*)
  Calcule les réactions au niveau des connexions ou des conditions aux limites.

Méthodes d'objet



Méthode AbsoluteId	Méthode AttributeType	Méthode GetAttributeBoolean
Méthode GetAttributeInteger	Méthode GetAttributeReal	Méthode GetAttributeString
Méthode HasAttribute	Méthode Id	Méthode IsOwnedBy
Méthode IsOwnedByString	IsSupporting	Méthode Name
Méthode SetAttributeBoolean	Méthode SetAttributeReal	Méthode SetAttributeInteger
Méthode SetAttributeReal	Méthode SetAttributeString	SetAttributeDimension

Méthode AbsoluteId

Extrait le chemin d'une entité. Notez que cette fonction n'est pas NLS (elle ne sera pas traduite dans votre langue, mais reste en anglais).

Syntaxe

feature.**AbsoluteId**(): String

Exemple

```
String.2=PartBody\Pad.1.Id() + PartBody\Pad.1.AbsoluteId()
```

Modèle

KwrTopology.CATPart

Méthode AttributeType

Renvoie le type d'attribut sous la forme d'une chaîne.

Méthode GetAttributeBoolean

Renvoie la valeur d'un paramètre de type booléen ajouté à une entité donnée en utilisant le produit

Knowledge Advisor. *parameterName* est le nom d'un paramètre de type booléen. Il doit être placé entre guillemets ("). Cette méthode permet de lire les éléments suivants :

- Les attributs ajoutés aux paramètres à l'aide de l'explorateur de paramètres.
- Les attributs réels ajoutés aux objets.
- Les propriétés utilisateur d'un produit.

Syntaxe

feature.**GetAttributeBoolean(***String***):** Boolean

où l'argument est le nom de l'attribut.

Exemple

```
Message ("La valeur de l'attribut Boolean.1 de # est #",
PartBody\Pad.1.Name(),
PartBody\Pad.1.GetAttributeBoolean("Boolean.1"))
```

Modèle

KwrObject.CATPart

Méthode GetAttributeInteger



Renvoie la valeur d'un paramètre de type entier ajouté à une entité donnée en utilisant le produit Knowledge Advisor. *parameterName* est le nom d'un paramètre de type chaîne. Il doit être placé entre guillemets ("). Cette méthode permet de lire les éléments suivants :

- Les attributs ajoutés aux paramètres à l'aide de l'explorateur de paramètres.
- Les attributs réels ajoutés aux objets.
- Les propriétés utilisateur d'un produit.

Syntaxe

feature.**GetAttributeInteger(***String***):** Integer

où *String* est le nom de l'attribut. Ce nom doit être placé entre guillemets doubles.

Exemple

```
Integer.3=PartBody\Hole.1 .GetAttributeInteger("Integer.2")
```

Modèle

KwrObject.CATPart

Méthode GetAttributeReal

Renvoie la valeur d'un paramètre de type réel ou longueur (en m) ajouté à une entité donnée en utilisant le produit Knowledge Advisor. *parameterName* est le nom d'un paramètre de type chaîne. Il doit être placé entre guillemets ("). Cette méthode permet de lire les éléments suivants :

- Les attributs ajoutés aux paramètres à l'aide de l'explorateur de paramètres.
- Les attributs réels ajoutés aux objets.
- Les propriétés utilisateur d'un produit.

Syntaxe

feature.**GetAttributeReal(***String***):** String

où *String* est le nom de l'attribut. Ce nom doit être placé entre guillemets doubles.

Méthode GetAttributeString

Renvoie la valeur d'un paramètre de type chaîne ajouté à une entité donnée en utilisant le produit Knowledge Advisor. *parameterName* est le nom d'un paramètre de type chaîne. Cette méthode permet de lire les éléments suivants :

- Les attributs ajoutés aux paramètres à l'aide de l'explorateur de paramètres.
- Les attributs réels ajoutés aux objets.
- Les propriétés utilisateur d'un produit.

Syntaxe

feature.**GetAttributeString(***String***):** String

où *String* est le nom de l'attribut. Ce nom doit être placé entre guillemets doubles.

Exemple

```
String.2 =PartBody\Pad.1 .GetAttributeString("String.1")
```

Modèle

KwrObject.CATPart

Méthode HasAttribute

Détermine si l'attribut spécifié dans l'argument appartient à l'entité à laquelle est appliquée la méthode.

Syntaxe

feature.**HasAttribute(***String***):** Boolean

où *String* est le nom de l'attribut. Ce nom doit être placé entre guillemets doubles.

Exemple

```
Boolean.2 =
PartBody\Hole.1.HasAttribute("Real.1")
```

Modèle

KwrObject.CATPart

Méthode Id

S'applique à une entité. Extrait l'identificateur d'une entité.

Syntaxe

feature.**Id**(): String

Exemple

```
String.2=PartBody\Pad.1.Id() + PartBody\Pad.1.AbsoluteId()
```

Modèle

KwrTopology.CATPart

Méthode IsOwnedBy

Détermine si l'attribut spécifié par l'argument est parent de l'entité à laquelle est appliquée la

méthode. *nom_entité* doit être entre guillemets (").

Syntaxe

feature.**IsOwnedBy**(): Boolean

Exemple

```
Boolean.1=PartBody\Hole.1.IsOwnedBy(PartBody)
```

Modèle

Topology.CATPart

Méthode IsOwnedByString



S'applique à une entité. Détermine si une entité appartient à une autre. Cette méthode renvoie une chaîne.

Syntaxe

feature.**IsOwnedByString**(): Boolean

IsSupporting



Fonction indiquant si l'objet transmis par un argument est pris en charge ou non.

Exemple

```
H:Hole
H->IsSupporting("TaperedHole") == true
```

Méthode Name



S'applique à une entité. Extrait le nom d'une entité. Ne peut pas s'utiliser pour renommer une entité.

Syntaxe

feature.**Name**(): String

Exemple

```
String.1=PartBody\Pad.1.Name()
```

Modèle

KwrTopology.CATPart

Méthode SetAttributeBoolean



Affecte la valeur spécifiée par le deuxième argument au paramètre dont le nom est indiqué par le premier argument. *parameterName* est le nom d'un paramètre de type booléen dont la valeur doit être modifiée. Il doit être placé entre guillemets ("). *valeur_booléen* a la valeur TRUE ou FALSE.

Syntaxe

feature.**SetAttributeBoolean(***String*, *Boolean***):** Void

où le premier argument est le nom de l'attribut tandis que le second est la valeur à lui affecter.

Exemple

```
if PartBody\Pad.1\Boolean.1 <> true
PartBody\Pad.1.SetAttributeBoolean("Boolean.1", true)
```

Modèle

KwrObject.CATPart

Méthode SetAttributeReal



Affecte la valeur spécifiée par le deuxième argument au paramètre dont le nom est indiqué par le premier argument. *parameterName* est le nom d'un paramètre de type réel dont la valeur doit être modifiée. *parameterName* doit être entre guillemets (").

Syntaxe

feature.**SetAttributeReal(***String*, *Real***):** Void

où *String* est le nom de l'attribut et *Real* la valeur à affecter au paramètre.

Exemple

```
if PartBody\Hole.1\Real.1 <> 3
PartBody\Hole.1 .SetAttributeReal("Real.1",3)
```

Modèle

KwrObject.CATPart

Méthode SetAttributeString

Affecte la valeur spécifiée par le deuxième argument au paramètre dont le nom est indiqué par le premier argument. *parameterName* est le nom d'un paramètre de type chaîne dont la valeur doit être modifiée. *parameterName* et *stringvalue* doivent être entre guillemets (").

Syntaxe

feature.**SetAttributeString(***String*, *String***):** Void

où le premier argument est le nom de l'attribut tandis que le second est la valeur à lui affecter.

Exemple

```
if PartBody\Pad.1.GetAttributeString("String.1") <> "String1"
PartBody\Pad.1 .SetAttributeString("String.1","This is a test")
```

La syntaxe suivante est également valable pour la même règle :

```
if PartBody\Pad.1\String.1 <> "String1"
PartBody\Pad.1.SetAttributeString("String.1","Il s'agit d'un test")
```

Modèle

KwrObject.CATPart

Méthode SetAttributeInteger

Affecte la valeur spécifiée par le deuxième argument au paramètre dont le nom est indiqué par le premier argument. *parameterName* est le nom d'un paramètre de type entier dont la valeur doit être modifiée. *parameterName* doit être entre guillemets (").

Syntaxe

feature.**SetAttributeInteger(***String*, *Integer***):** Void

où le premier argument est le nom de l'attribut tandis que le second est la valeur à lui affecter.

Exemple

```
if PartBody\Hole.1\Integer.1 <> 3
PartBody\Hole.1 .SetAttributeInteger("Integer.1", 3)
```

Modèle

KwrObject.CATPart

Méthode SetAttributeDimension

Affecte la valeur spécifiée par le deuxième argument au paramètre dont le nom est indiqué par le premier argument. *parameterName* est le nom du paramètre dont la valeur doit être modifiée. *parameterName* doit être placé entre guillemets ("). L'unité utilisée est l'unité courante.

Syntaxe

feature.**SetAttributeDimension(***String, Real, String***):** Void Type

où le premier argument est le nom de l'attribut, le deuxième est la valeur à lui affecter et le troisième le type.

Exemple

```
H:Hole

H->SetAttributeDimension("Myhole",5,"Length")
```

Fonctions mathématiques

Modèle (illustre les interpolations) : KwrInterpolations.CATPart

Notez que l'entrée de la fonction cosinus doit être un angle, même si sa signature autorise des nombres réels. Ainsi cos(3.14) est équivalent à cos (3.14 * 1rad) dans l'éditeur de formules. Toutefois, un avertissement s'affiche. Notez que cette affirmation s'applique également aux fonctions sinus et tangent.

- **abs**(Real): Real
  Calcule la valeur absolue d'un nombre.
- **ceil**(Real): Real
  Renvoie la plus petite valeur entière supérieure ou égale à la valeur définie par l'argument.
- **floor**(Real):Real
  Renvoie la plus grande valeur entière supérieure ou égale à la valeur définie par l'argument.
- **int**(Real):Real
  Renvoie la valeur entière d'un nombre.
- **let**
  Affecte une valeur à une variable temporaire (let x = 30 mm).
- **min**(Real,Real):Real, **max**(Real,Real)
  Renvoie la valeur minimale ou maximale d'une série de valeurs définie par l'argument.
- **sqrt**(Real):Real
  Renvoie la racine carrée.
- **log**(Real):Real
  Renvoie le logarithme.
- **ln**(Real):Real
  Renvoie le logarithme népérien.
- **round**(Real):Real
  Renvoie un nombre arrondi.
- **round**(Real, String, Integer):Real
  Renvoie un nombre arrondi.
  - For Real = 13.552mm
  - String = m (for meter)
  - Integer = 2

  Le nombre renvoyé est 13 mm

- **exp**(Real):Real
  Renvoie l'exponentielle.

- **LinearInterpolation**(*arg1*:Real, *arg2*:Real, *arg3*:Real) : Real
  A utiliser lors de la création d'une courbe parallèle à partir d'une loi.
  Exemple :
  1 - Créez une droite dans l'atelier Generative Shape Design.
  2 - Ouvrez l'atelier Knowledge Advisor et créez la loi suivante :
  FormalReal.1 = LinearInterpolation(1,9,FormalReal.2)
  3 - Retournez dans l'atelier Generative Shape Design et créez une courbe parallèle. Sélectionnez le mode Loi et définissez la loi ci-dessus comme étant celle à appliquer.

- **CubicInterpolation**(*arg1*:Real, *arg2*:Real, *arg3*:Real) : Real
  A utiliser lors de la création d'une courbe parallèle à partir d'une loi.
  Exemple :
  1 - Créez une droite dans l'atelier Generative Shape Design.
  2 - Ouvrez l'atelier Knowledge Advisor et créez la loi suivante :
  FormalReal.1 = CubicInterpolation(1,50,FormalReal.2)
  3 - Retournez dans l'atelier Generative Shape Design et créez une courbe parallèle. Sélectionnez le mode Loi et définissez la loi ci-dessus comme étant celle à appliquer.

- **mod(Real,Integer): Real**
  Permet à l'utilisateur d'extraire le reste de la partie entière du réel divisée par l'entier.
  Syntaxe : mod(Real,Integer):Real
  Modèle : Mod.CATPart

- **Cos**(Real):Real**, cosh** (Real): Real
  Calcule le cosinus (cos) ou le cosinus hyperbolique (cosh).
  Exemple
  Real.1 = cos(PI*1rad/4)
  Real.1 = cos(45deg)

- **tan**(Real): Real, **tanh**(Real): Real
  Calcule la tangente (tan) ou la tangente hyperbolique (tanh).

- **sin**(Real):Real, **sinh**(Real):Real
  Calcule le sinus ou le sinus hyperbolique.

- **asin**(Real):Real, **asinh**(Real):Real
  Calcule l'arcsinus ou l'arcsinus hyperbolique.

- **acos**(Real):Real, **acosh**(Real):Real
  Calcule l'arccosinus ou l'arccosinus hyperbolique.

- **atan**(Real):Real, **atanh**(Real):Real
  Calcule l'arctangente ou l'arctangente hyperbolique.

- **max**(arg1: Real, arg2: Real, ...):Real
  Renvoie le paramètre le plus élevé.

- **min**(arg1: Real, arg2: Real, ...):Real
  Renvoie le paramètre le plus faible.

- **not**(Boolean):Boolean
  Vérifie le contraire de l'expression entre parenthèses.
  Exemple : not(A==B) est équivalent à A<>B.



Pour que ces méthodes soient efficaces, vous devez utiliser uniquement des nombres réels.

Analyse 3D

ClashOrContact	DistanceMin
DistanceMinXYZ	IsIncludedIn
PenetrationMax	

ClashOrContact

Syntaxe

ClashOrContact(*p1: String,Product,Product,...*):**Boolean**
Détermine si deux composants sont en collision ou en contact. Le premier argument est "Clash" ou "Contact" suivant le type d'analyse désiré.

DistanceMin

Syntaxe

DistanceMin(*p1: Product, p2:Product,...***):Length**
Renvoie la distance minimale entre deux composants.

DistanceMinXYZ

Syntaxe

DistanceMin(*p2: String, p1:Product, p2: Product, ...***):Length**
Renvoie la distance minimale le long d'une direction entre deux composants. Le premier argument est "X", "Y" ou "Z" en fonction de la direction.

IsIncludedIn

Syntaxe

IsIncludedIn(*Product1***,***Product2***):Boolean**
Détermine si un composant est inclus dans un autre.



PenetrationMax

Syntaxe

PenetrationMax(*p1: Product, p2:Product,...***):Length**
Renvoie la longueur maximale d'un composant inclus dans un autre.



Création d'une formule

Dans cette tâche, vous apprendrez à créer une formule qui indique que le rayon externe d'un cylindre creux correspond au double de son diamètre interne. Notez que le rayon d'une esquisse peut se définir par une formule, du moment que cette dernière est déclarée comme contrainte.

Assurez-vous que l'option Relations est activée dans l'onglet **Outils -> Options... -> Infrastructure -> Infrastructure Part -> Affichage**.

1. Ouvrez le document KwrStartDocument.CATPart.
2. Cliquez sur l'icône  pour afficher la boîte de dialogue *f(x)*. Vérifiez que l'option Incrémental est désélectionnée.

Méthode 1

- Cliquez deux fois sur le paramètre Corps_Principal\Esquisse.1\Rayon.3\Rayon dans la liste des paramètres. L'éditeur de formules s'affiche.
- Entrez la relation **Corps_Principal\Trou.1\HoleLimit.1\Profondeur*2** dans le champ d'entrée des formules. Pour plus d'informations sur la manipulation des paramètres et des formules, reportez-vous à la section Conseils et astuces.
- Cliquez sur **OK** dans l'éditeur de formules.

Méthode 2

- Sélectionnez le paramètre Corps_Principal\Esquisse.1\Rayon.1\Rayon dans la liste des paramètres.

- Cliquez sur **Ajouter formule**. L'éditeur de formules s'affiche.

- Entrez la relation **Corps_Principal\Trou.1\HoleLimit.1\Profondeur*2** dans le champ d'entrée des formules. Pour plus d'informations sur la manipulation des paramètres et des formules, reportez-vous à la section Conseils et astuces.

- Cliquez sur **OK** dans l'éditeur de formules.

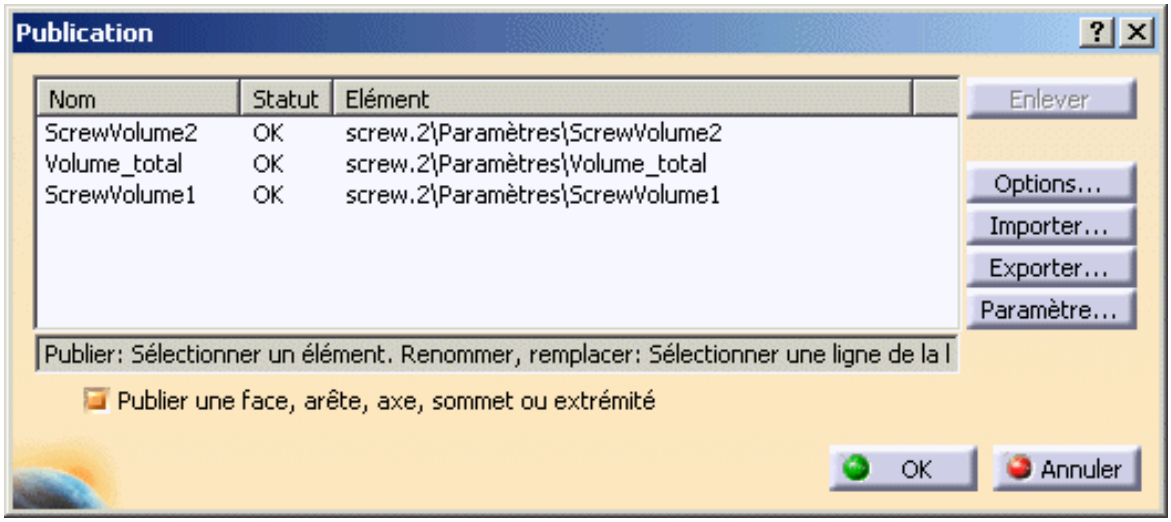
3. Cliquez sur **Appliquer** pour mettre à jour le document.
4. Cliquez sur **OK** pour fermer la boîte de dialogue.



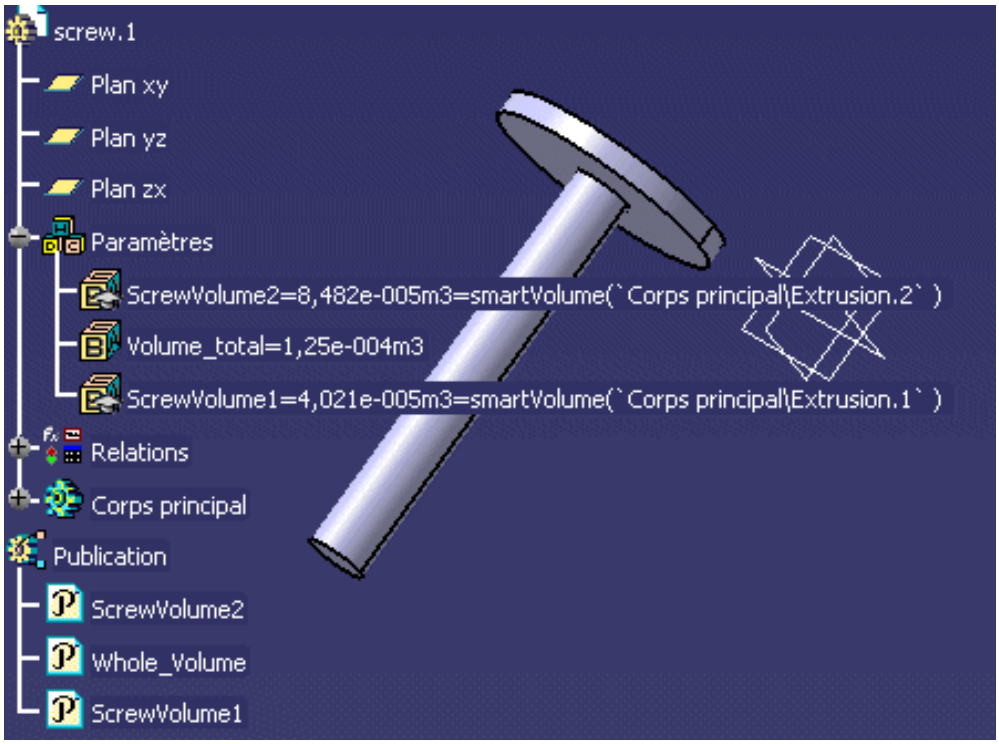
Création de formules basées sur des publications

Dans cette tâche, vous apprendrez à créer une formule basée sur les publications d'un fichier CATProduct.

1. Ouvrez le document Screw1.CATPart.
2. Ajoutez un paramètre de volume à la pièce. Pour ce faire, procédez comme suit :
   - Cliquez sur l'icône f(x). L'éditeur de formules s'ouvre.Dans la liste déroulante **Créer paramètre de type**, sélectionnez **Volume**, puis cliquez sur le bouton **Créer paramètre de type**.
   - Dans le champ **Editer le nom ou la valeur du paramètre sélectionné**, entrez le nom du paramètre : **ScrewVolume1**. Cliquez sur **Appliquer**, puis sur le bouton **Ajouter formule**. L'éditeur de formules s'ouvre.
   - Entrez la formule suivante à l'aide du dictionnaire pour accéder à l'opérateur smartVolume : **ScrewVolume1=smartVolume(PartBody\Pad.1)**. Cliquez trois fois sur **OK**.
3. Créez un autre paramètre Volume appelé ScrewVolume2 basé sur Pad.2.
4. Créez un autre paramètre Volume appelé Volume_Total. Lorsque vous avez terminé, cliquez sur **OK** pour quitter l'éditeur de formules.
5. Ouvrez le menu **Outils -> Publication** et sélectionnez les trois paramètres qui s'affichent sous le noeud Paramètres. Affectez-leur de nouveaux noms (voir graphique ci-après) :

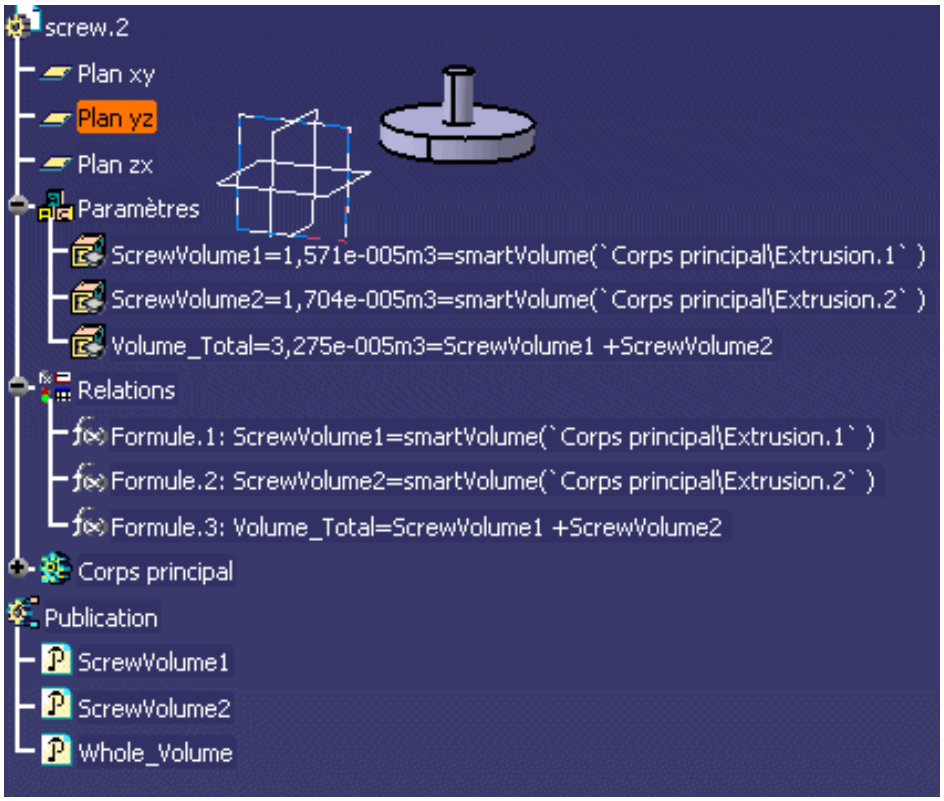


6. Lorsque vous avez terminé, cliquez sur **OK**. Les paramètres, les formules et les publications sont créés (voir graphique ci-après).



7. Enregistrez le fichier, puis fermez-le.

8. Ouvrez le document Screw2.CATPart. Répétez les étapes précédentes (2 à 7).



9. Créez un produit (menu **Fichier -> Nouveau**). Lorsque vous avez terminé, cliquez sur **OK**.

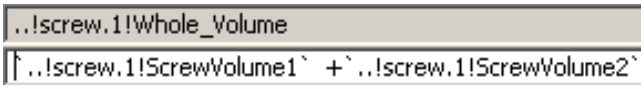
10. Choisissez la commande **Insertion -> Composant existant...** pour insérer Screw1.CATPart dans le produit.

11. Dans la fenêtre Sélection de fichier, sélectionnez le fichier Screw1.CATPart que vous venez d'enregistrer. Cliquez sur **Ouvrir**.

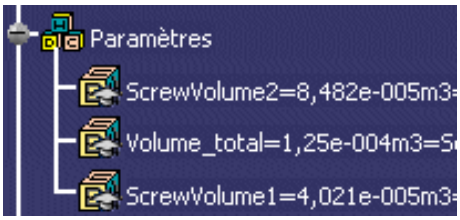
12. Créez une formule pour calculer le volume de la vis. Pour ce faire, procédez comme suit :

    - Cliquez sur le produit racine et cliquez sur l'icône  . L'éditeur de formules s'ouvre.

    - Entrez la formule suivante dans l'éditeur en sélectionnant les publications dans l'arbre de spécifications :

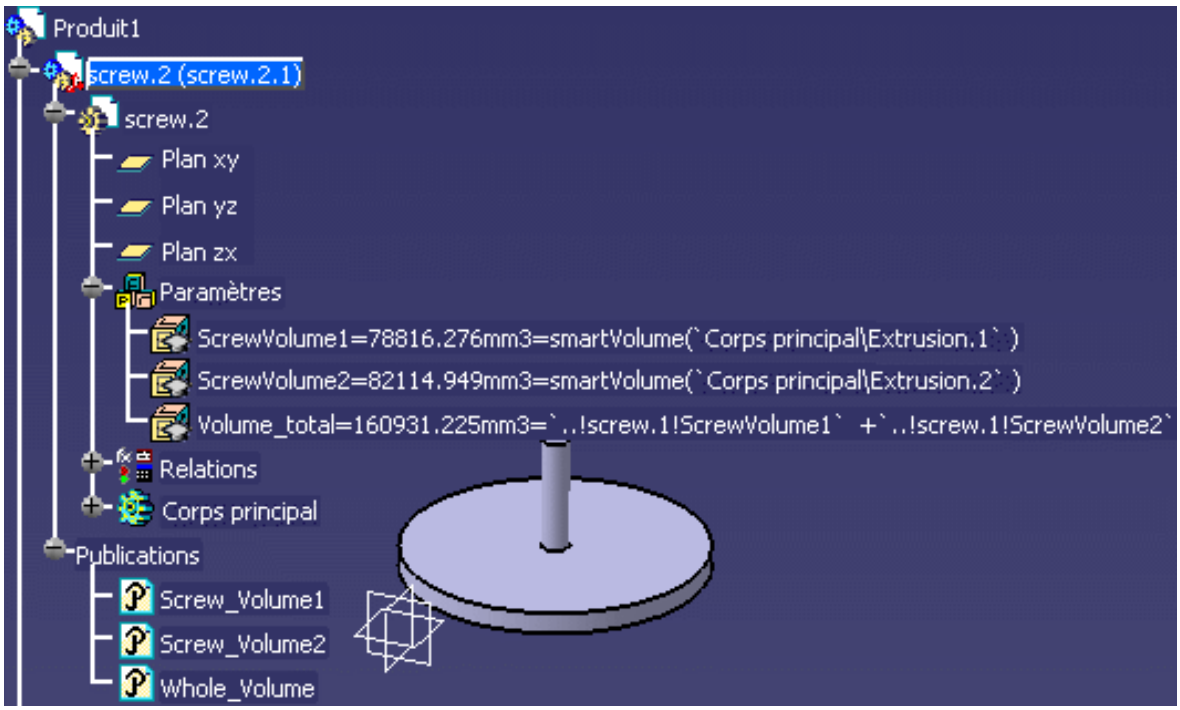
    - Lorsque vous avez terminé, cliquez sur **OK**. Le volume de la vis s'affiche sous le noeud Paramètres.

13. Cliquez deux fois sur screw.1 dans l'arbre de spécifications, cliquez dessus à l'aide du bouton droit de la souris et sélectionnez la commande **Composants -> Remplacer le composant**.

14. Dans la fenêtre Sélection de fichier, sélectionnez le fichier screw2.CATPart que vous venez d'enregistrer et cliquez sur **Ouvrir**.

15. Cliquez sur **OK** dans la fenêtre Les impacts au remplacement de ce produit. La nouvelle vis est insérée dans le produit et son volume est calculé.



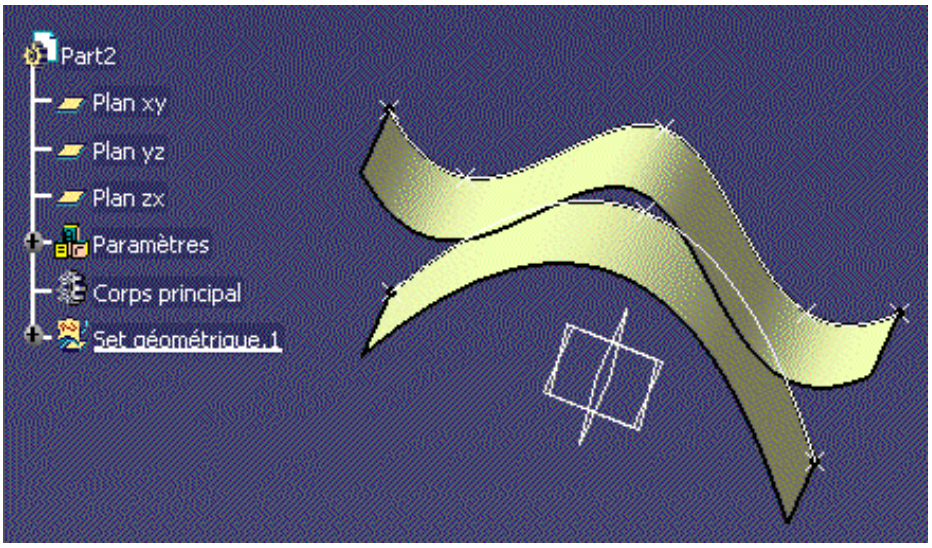
Définition d'une entité de mesure dans une formule

Dans cette tâche, vous apprendrez à indiquer que la valeur d'un paramètre de type Longueur est égale à l'abscisse curviligne d'un point situé sur une courbe.

Vous pouvez utiliser dans les formules des mesures, c'est-à-dire des valeurs capturées dans la zone géométrique. Voici quelques exemples de mesures susceptibles d'être utilisées dans des formules :

- Distance entre deux points.
- Longueur totale d'une courbe.
- Longueur d'un segment courbe - entre un point et son origine, ou entre un point et l'extrémité de la courbe.
- Longueur d'un segment courbe - entre deux points.
- Zone d'une surface extrudée.

1. Cochez la case **Chargement des bibliothèques d'extension du langage** dans l'onglet **Outils -> Options -> Général -> Paramètres et mesure -> langage**.
2. Ouvrez le document KwrMeasure.CATPart . Tout le document a été créé à l'aide du produit Generative Shape Design. Les surfaces Extrude.1 et Extrude.2 sont extrudées des courbes Spline.1 et Spline.2. Le point dont l'abscisse doit être mesurée est le Point.5. L'origine de la courbe sur laquelle est située le Point.5 est le Point.8.



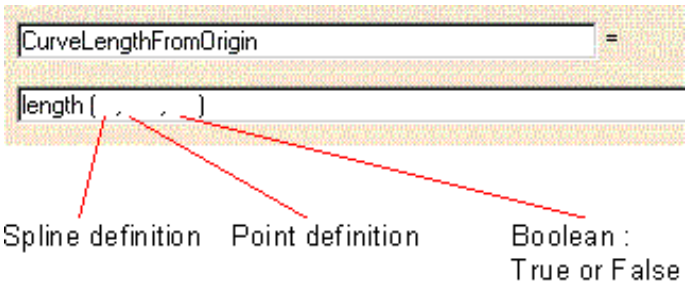
3. Cliquez sur l'icône Formule. La boîte de dialogue *f(x)* s'affiche.
4. Créez le paramètre CurveLengthFromOrigin. Pour ce faire, procédez comme suit :
   - Sélectionnez le composant Longueur avec Simple valeur dans la liste Créer paramètre de type, puis cliquez sur Créer paramètre de type. Le nouveau paramètre apparaît dans le champ Editer le nom ou la valeur du paramètre sélectionné.
   - Remplacez le nom Distance.1 par CurveLengthFromOrigin, puis cliquez sur Appliquer.

5. Spécifiez que la valeur de CurveLengthFromOrigin est l'abscisse du Point.5 :

   a. Sélectionnez le paramètre CurveLengthFromOrigin dans la liste des paramètres, puis cliquez sur Ajouter formule. L'éditeur de formules s'affiche.

   b. Sélectionnez le composant Mesures dans la liste du dictionnaire.

   c. Dans la liste des mesures, cliquez deux fois sur le composant de longueur (Courbe,Point,Booléen). La fonction de longueur est ajoutée à l'éditeur de formules.



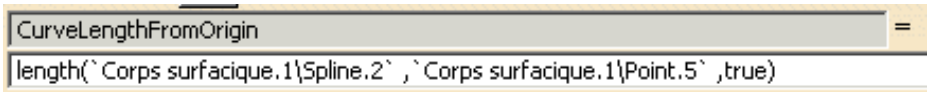
   d. Complétez le champ de l'éditeur de formules comme indiqué ci-après.

      1. Les trois arguments sont les suivants : une courbe à sélectionner à partir dans la zone géométrique, un point à sélectionner dans la zone géométrique et un opérateur booléen.

      2. Positionnez le curseur à l'emplacement prévu pour le premier argument. Ensuite, cliquez deux fois sur le composant Spline.2 dans l'arbre de spécifications. L'argument de courbe est ajouté à la définition de la longueur.

      3. Positionnez le curseur à l'emplacement prévu pour le deuxième argument. Ensuite, cliquez deux fois sur le composant Point.5 dans l'arbre de spécifications. L'argument de point est ajouté à la définition de la longueur.

      4. Saisissez un opérateur booléen pour le troisième argument : Vrai si la longueur doit être calculée à partir de l'origine, Faux si la longueur doit être calculée à partir de l'extrémité de la courbe.



      5. Cliquez sur **OK** pour confirmer la définition de la formule. Vous revenez dans la boîte de dialogue Formules. La formule CurveLengthFromOrigin et la valeur (47,5 mm) sont ajoutées dans la liste des paramètres.

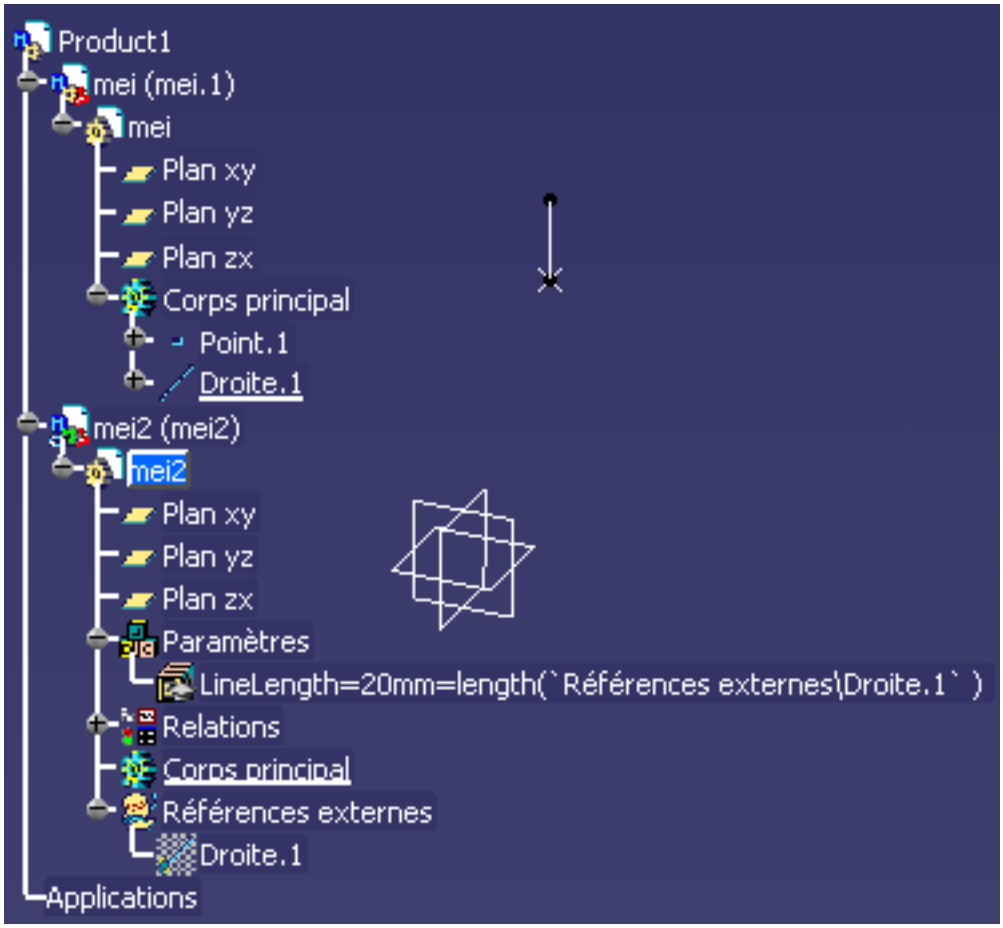
   e. Cliquez sur **OK** pour ajouter le paramètre ainsi que sa formule dans le document.

Création d'une formule à l'aide d'éléments géométriques

Dans cette tâche, vous apprendrez à créer une formule à l'aide de la géométrie.

Avant d'exécuter ce scénario, assurez-vous que l'option **Garder le lien avec l'objet sélectionné** est activée (**Outils -> Options -> Infrastructure -> Part Infrastructure -> Général**).

1. Créez un produit et ajoutez-lui 2 pièces.
2. Dans le menu **Démarrer -> Forme**, ouvrez l'atelier **Generative Shape Design**. Sélectionnez la première pièce.
3. Ajoutez une droite à la pièce.
4. Sélectionnez la deuxième pièce et ajoutez-lui un paramètre Longueur ainsi qu'une formule basée sur la droite. Pour ce faire, procédez comme suit :
   - Cliquez sur l'icône  . L'éditeur de formules s'ouvre.Dans la liste déroulante **Créer paramètre de type**, sélectionnez **Longueur**, puis cliquez sur le bouton **Créer paramètre de type**.
   - Dans le champ **Editer le nom ou la valeur du paramètre sélectionné**, entrez le nom du paramètre : **LineLength**. Cliquez sur **Appliquer**, puis sur le bouton **Ajouter formule**. L'éditeur de formules s'ouvre.
   - Entrez la formule suivante à l'aide du dictionnaire pour accéder à l'opérateur de longueur (Courbe,...) : LineLength=length().
   - Placez le curseur entre parenthèses et cliquez deux fois sur la droite ou dans l'arbre de spécifications. La fenêtre Sélection de paramètre externe s'affiche.
   - Sélectionnez Line.1 et cliquez sur **OK**. La formule s'affiche comme suit : LineLength=length(`External References\Line.1`)
   - Lorsque vous avez terminé, cliquez sur **OK**.Cliquez sur **Oui** dans la fenêtre Mise à jour automatique.


Product1
mei (mei.1)
mei
Plan xy
Plan yz
Plan zx
Corps principal
Point.1
Droite.1
mei2 (mei2)
mei2
Plan xy
Plan yz
Plan zx
Paramètres
LineLength=20mm=length(`Références externes\Droite.1`)
Relations
Corps principal
Références externes
Droite.1
Applications

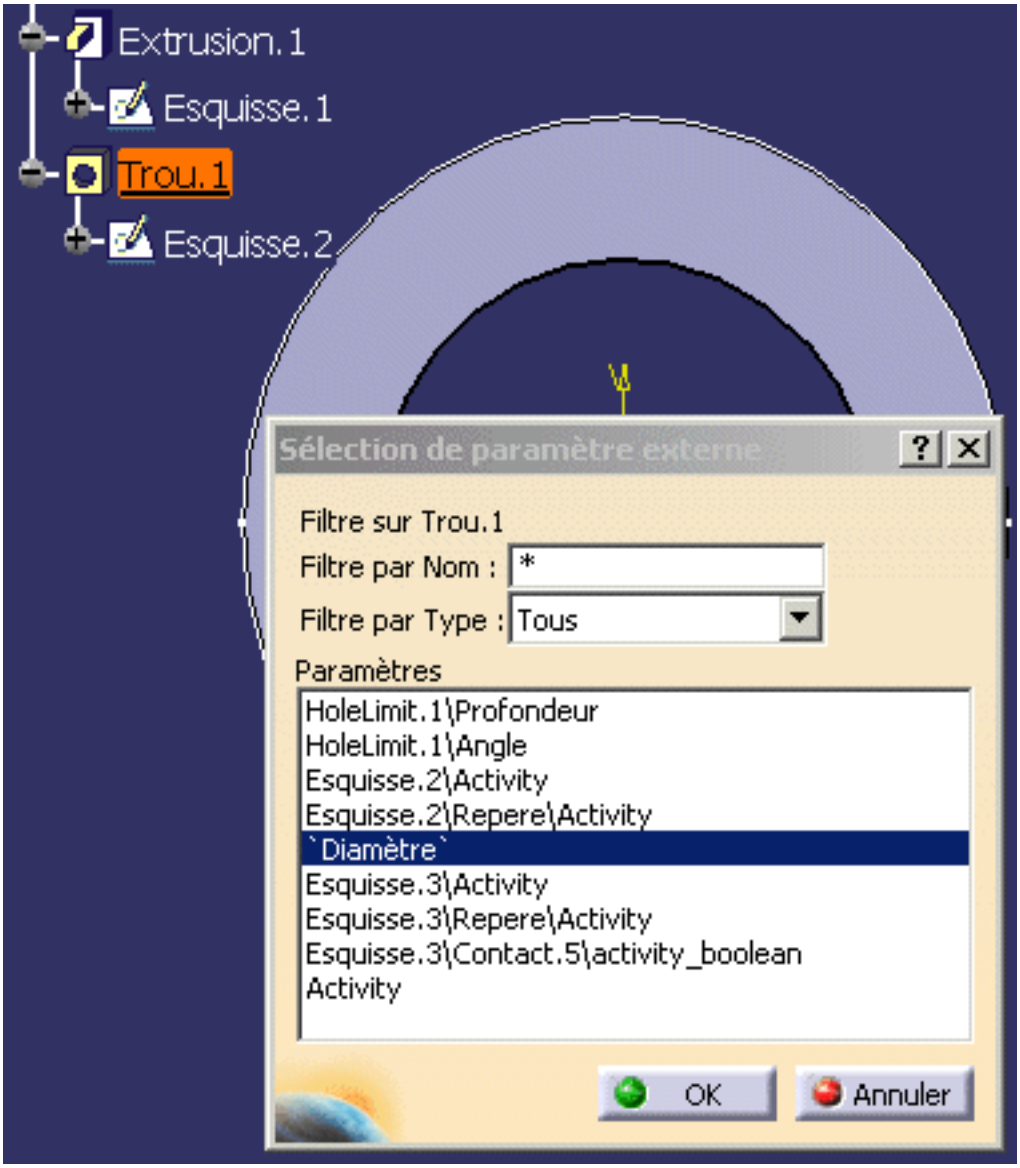
Référence à des paramètres externes dans une formule

Ce scénario indique comment utiliser des paramètres externes dans une formule.

Dans une formule, vous pouvez utiliser des paramètres définis dans des documents externes, quel que soit le type de document. Par exemple, dans un document CATPart, vous pouvez spécifier une formule faisant référence aux paramètres définis dans un document CATDrafting. Les paramètres externes peuvent également être utilisés avec un assemblage.

Avant d'exécuter ce scénario, assurez-vous que l'option **Garder le lien avec l'objet sélectionné** est activée (**Outils -> Options... -> Infrastructure -> Infrastructure Part -> Général**).

1. Ouvrez le document KwrStartDocument.CATPart ainsi que le document KwrImportParameter.CATPart. Sélectionnez la commande **Fenêtre -> Mosaïque verticale** dans la barre de menus standard. Les deux documents s'affichent.
2. Sélectionnez le document KwrImportParameter. Cliquez sur l'icône  pour afficher la boîte de dialogue *f(x)*.
3. Créez un paramètre de type Longueur et cliquez sur le bouton Ajouter formule. L'éditeur de formules s'affiche.
4. Dans l'arbre de spécifications KwrStartDocument, sélectionnez le composant Trou.1. La boîte de dialogue **Sélection de paramètre externe** s'affiche.



5. Dans la boîte de dialogue **Sélection de paramètre externe**, sélectionnez l'objet Diamètre dans la liste des paramètres externes. Cliquez ensuite sur OK. La définition Distance.1 est transmise à l'éditeur de formules. (Cliquez sur l'image ci-dessous pour l'agrandir.)

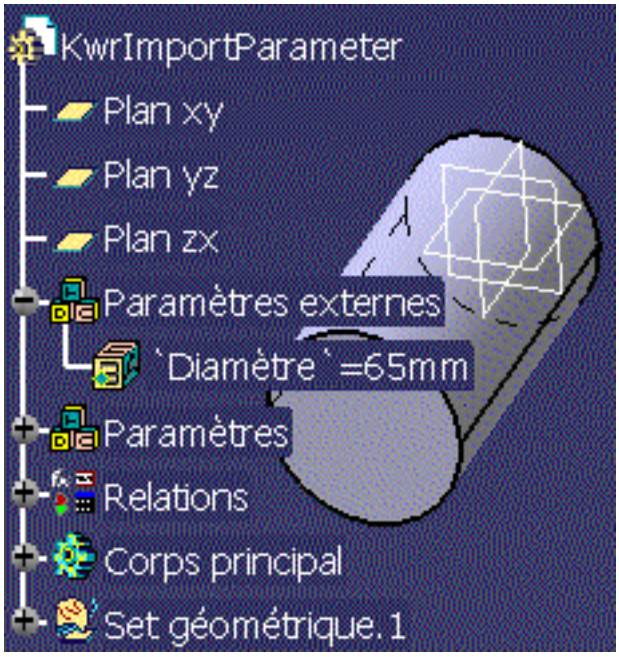


6. Complétez la définition de formule comme indiqué ci-après :

   Distance.1 = Diamètre*0,45

7. Cliquez sur **OK** dans l'éditeur de formules. Vous revenez dans la boîte de dialogue Formules. Dans la liste des paramètres, la valeur du paramètre

Distance.1 est modifiée en fonction de la formule spécifiée. Dans l'arbre de spécifications KwrImportParameter, le noeud Paramètres externes est ajouté. Développez ce noeud pour afficher le paramètre Diamètre.



8. Cliquez sur **OK** pour ajouter la formule dans le document KwrImportParameter.CATPart, puis quittez la boîte de dialogue.
9. Sélectionnez la commande Edition -> Liaisons dans la barre de menus standard. La boîte de dialogue affichée confirme qu'il existe une liaison entre l'objet KwrImportParameter\Longueur.1 et l'objet KwrStartDocument\PartBody\Trou.1\Diamètre.
10. Cliquez sur **Isoler** dans la boîte de dialogue Liaisons, puis sur **OK**. Dans l'arbre de spécifications KwrImportParameter, le noeud Paramètres externes ne peut plus être développé et le paramètre Diamètre est ajouté en aval du noeud Paramètres.



11. Sélectionnez la commande **Edition -> Liaisons** dans la barre de menus standard. Un message vous informe que le document actif ne comporte aucune liaison externe.

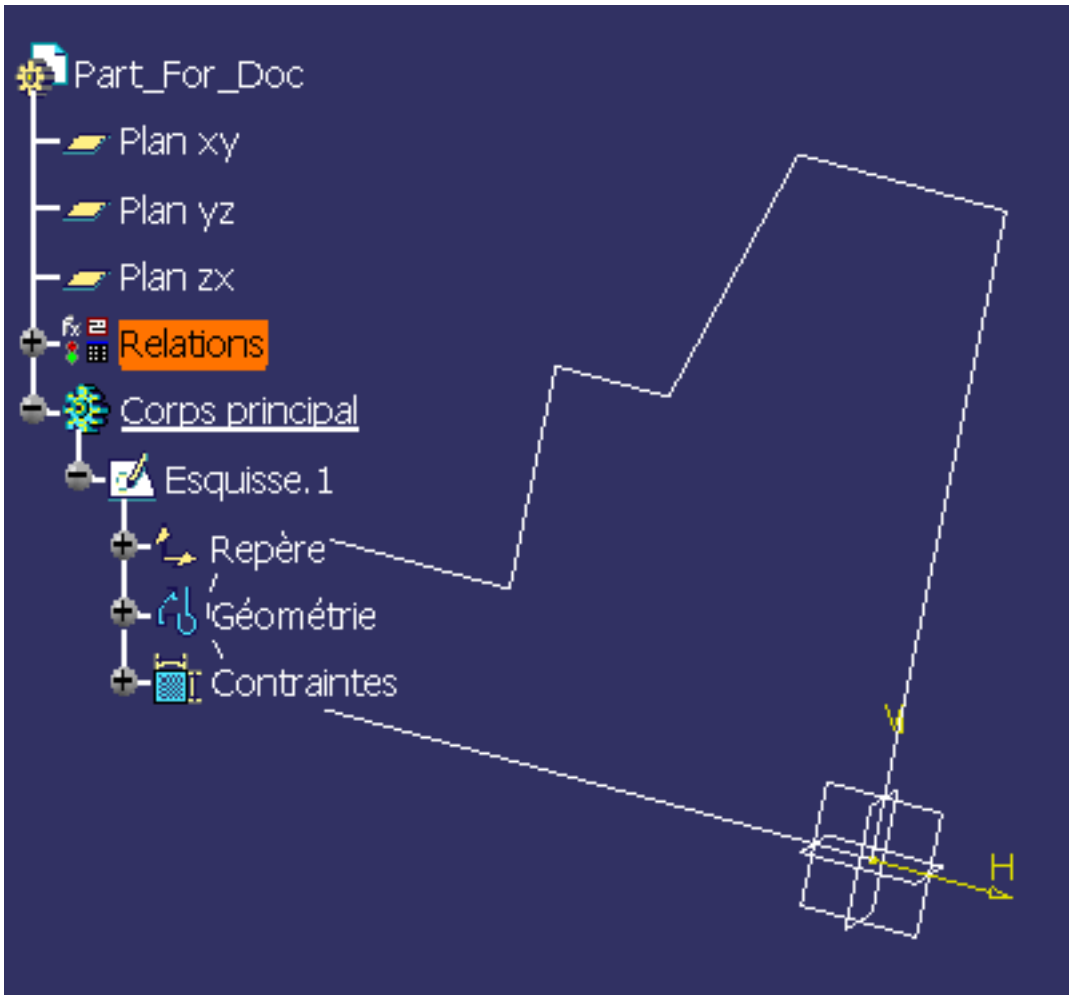
Utilisation de la fonction Equivalence

Dans cette tâche, vous apprendrez à utiliser l'équivalence. Le scénario ci-après se compose des étapes suivantes :

- L'utilisateur applique des contraintes à une esquisse existante.
- L'utilisateur utilise l'équivalence pour créer une liste de paramètres de type longueur qui auront la même valeur.

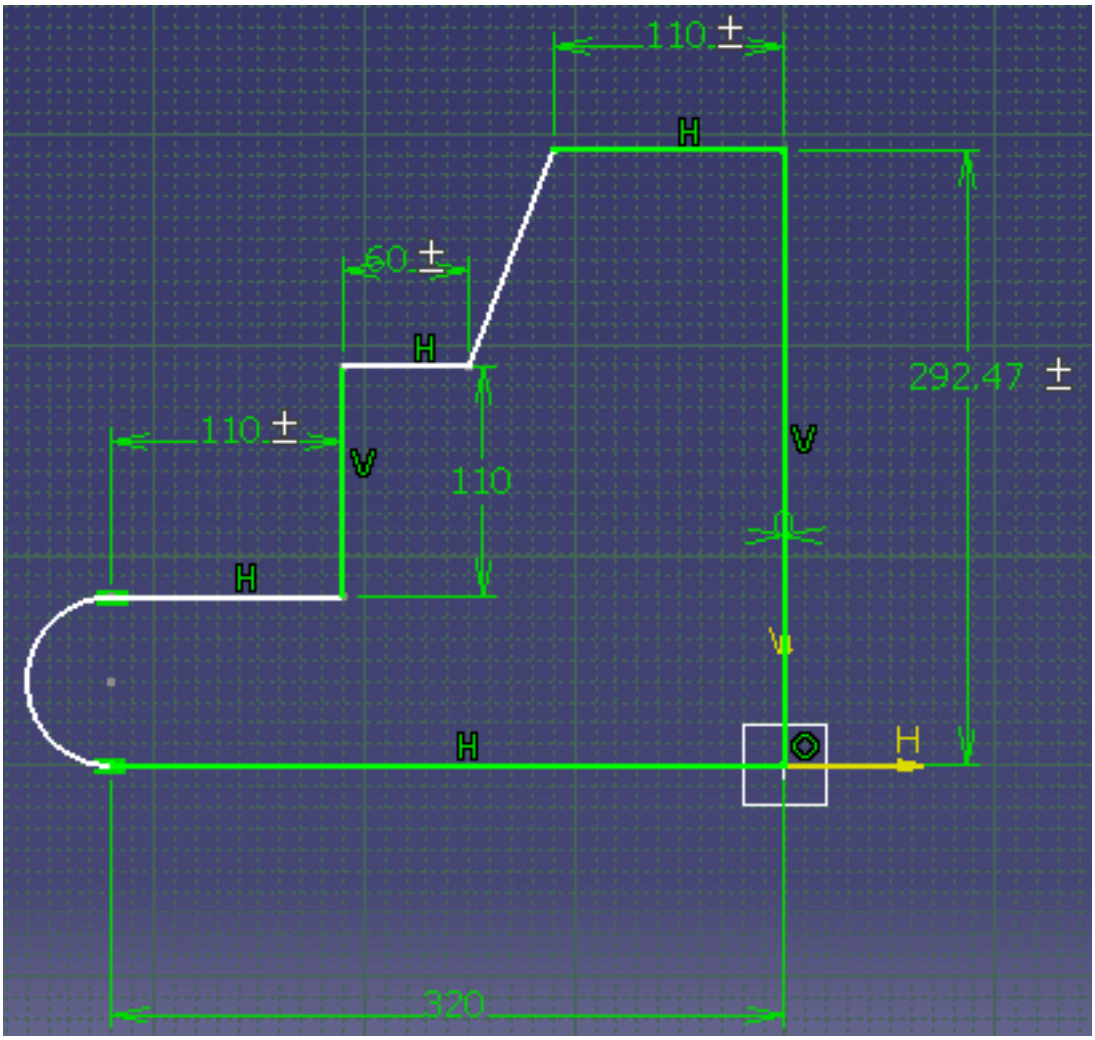
Pour en savoir plus sur l'équivalence, reportez-vous à la section Familiarisation avec l'interface d'équivalence.

1. Ouvrez KwrEquivalentDimensions.CATPart. L'image suivante s'affiche :



2. Dans l'arbre de spécifications, développez le noeud PartBody et cliquez deux fois sur Esquisse.1 pour accéder à l'atelier d'esquisse.

3. Cliquez deux fois sur l'icône **Contrainte** ( ) pour assortir certaines lignes de l'esquisse de contraintes (voir le graphique ci-dessous).



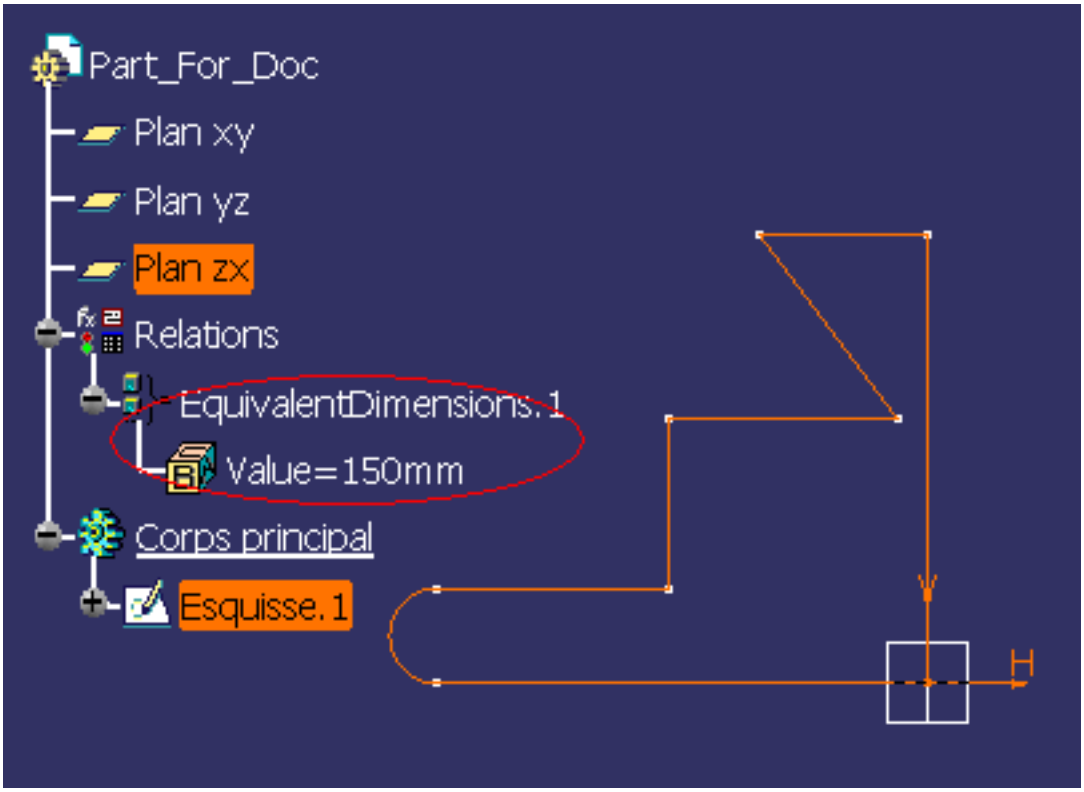
4. Dans la barre d'outils Knowledge, cliquez sur l'icône **Equivalence** (). La fenêtre Edition de l'équivalence s'affiche.

5. Cliquez sur le bouton **Edition de la liste...**. Dans la fenêtre qui apparaît, sélectionnez les paramètres suivants à l'aide de la touche de direction, puis cliquez sur **OK**.

   - Longueur.34
   - Longueur.36
   - Longueur.37

6. Dans la fenêtre Edition de l'équivalence, définissez la valeur sur 150 mm et cliquez sur **OK**.

7. Quittez l'atelier Sketcher. L'esquisse est modifiée en conséquence et le composant EquivalentDimensions.1 s'affiche sous le noeud Relations.



8. Cliquez deux fois sur Value=150 mm dans l'arbre de spécifications. La fenêtre **Editer paramètre** s'affiche.

9. Entrez 140 mm et cliquez sur **OK**.

Association d'adresses URL & de commentaires à des entités

Association d'adresses URL et de commentaires à des paramètres ou à des relations
Affectation d'un commentaire à une entité Knowledge

Association d'adresses URL & de commentaires à des paramètres ou à des relations

Vous pouvez associer une ou plusieurs adresses URL aux relations et aux paramètres utilisateur. Cette tâche s'avère utile uniquement lorsque le document actif contient des relations et/ou des paramètres utilisateur.

Ajout d'adresses URL

1. Cliquez sur l'icône **Commentaire & URL** () dans la barre d'outils Organisation de la connaissance. La boîte de dialogue **URL & commentaire** apparaît.
2. Dans l'arbre de spécifications, sélectionnez un paramètre ou une entité de type relation.
3. Sélectionnez l'onglet **Edition**, puis cliquez sur le bouton **Ajouter**. La boîte de dialogue **Ajouter un URL** s'affiche.
4. Entrez une adresse URL (http:\\www.foo.org par exemple) ainsi qu'un nom. Cliquez sur OK.
5. Le cas échéant, répétez cette opération pour ajouter une autre adresse URL.
6. Une fois les adresses URL requises entrées, ajoutez un commentaire et cliquez sur **OK** pour fermer la boîte de dialogue. Les adresses URL et le commentaire sont ajoutés à l'entité sélectionnée.

- Lorsque vous utilisez la boîte de dialogue Commentaire & URL, notez que l'icône  située dans la barre d'outils Knowledge Advisor permet à l'utilisateur expérimenté d'accéder à une boîte de dialogue Commentaire & URL dans laquelle vous pouvez ajouter, supprimer et modifier les URL. L'icône  disponible dans la barre d'outils générale est destinée à l'utilisateur final uniquement.

- Pour vérifier que des adresses URL ont été affectées à une relation ou à un paramètre utilisateur, il suffit de cliquer sur l'icône Commentaire & URL et de sélectionner l'objet qui convient dans l'arbre de spécifications.

Recherche d'une adresse URL

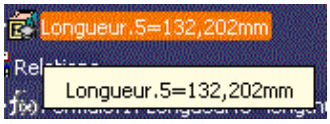
Lorsque plusieurs adresses URL ont été affectées à un objet, l'onglet Explorer de la boîte de dialogue URL & commentaire permet de rechercher une adresse URL en particulier.

1. Cliquez sur l'icône  et sélectionnez un objet (relation ou paramètre utilisateur) dans l'arbre de spécifications.
2. Sélectionnez l'onglet **Explorer**. La liste des adresses URL affectées à l'objet sélectionné apparaît.
3. Entrez le nom (ou une sous-chaîne) de l'URL à rechercher dans le champ Rechercher, puis cliquez sur **Rechercher**. Si l'adresse URL définie est trouvée, la mention "oui" apparaît dans la colonne Trouvé pour chaque objet contenant une adresse URL correspondant à la recherche et seul le premier objet trouvé est mis en évidence.

Affectation d'un commentaire à une entité Knowledge

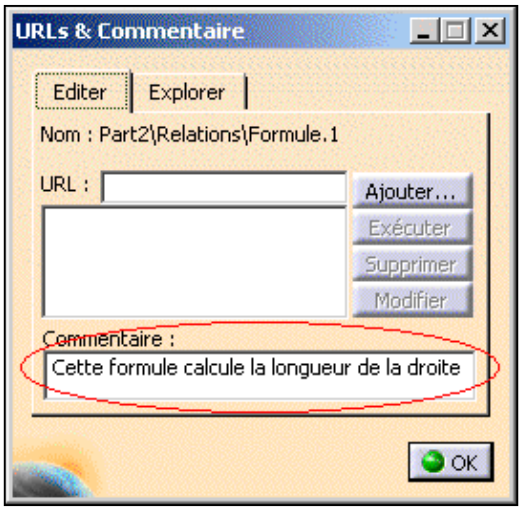
Dans cette tâche, vous apprendrez à ajouter des commentaires aux paramètres et aux formules à l'aide de la boîte à outils **Commentaire & URL** et de la boîte de dialogue **Propriétés**.

Vous pouvez maintenant modifier le texte qui apparaît dans les info-bulles lorsque vous faites glisser votre souris sur une relation ou un paramètre. Vous pouvez donc affecter un texte à afficher à une entité particulière.



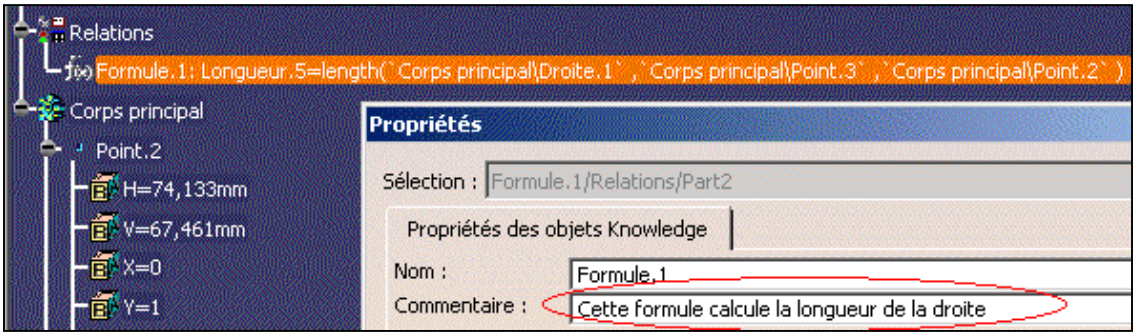
Vous pouvez modifier le texte des info-bulles de deux façons :

1. Sélectionnez l'entité à laquelle vous souhaitez ajouter un commentaire.
2. Cliquez sur l'icône **Commentaire & URL** ( ).
3. Entrez le texte que vous souhaitez afficher dans le champ Commentaire.
4. Cliquez sur **OK** lorsque vous avez terminé.



- Ou -

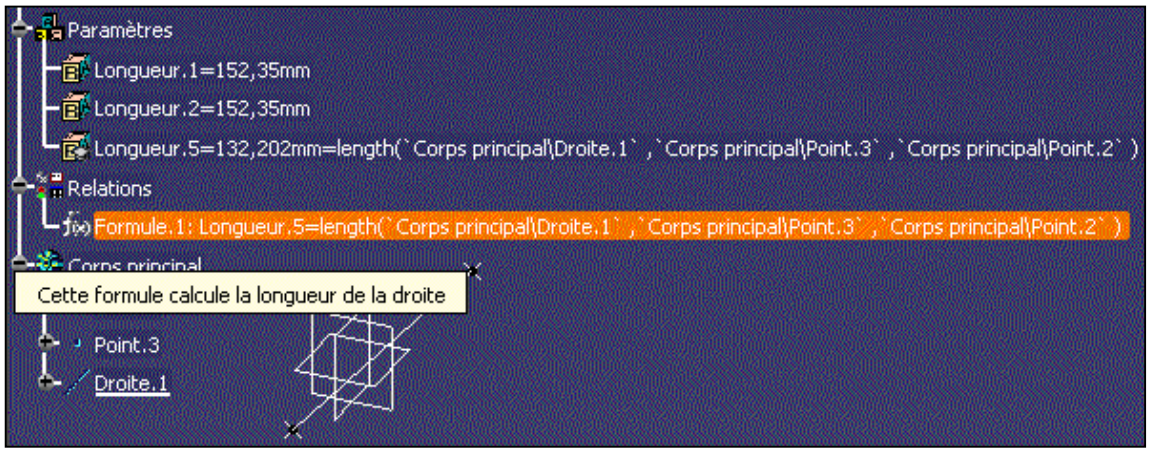
1. Cliquez avec le bouton droit de la souris sur la relation dans l'arbre de spécifications.
2. Sélectionnez la commande **Propriétés**.
3. Entrez le commentaire dans le champ **Commentaire** de la boîte de dialogue **Propriétés**.



Remarques :

- Pour afficher la valeur du paramètre/de la formule, vous devez utiliser la combinaison "**\V**" (voir l'exemple ci-dessous).
- Si aucun commentaire n'est ajouté à l'entité, le nom par défaut de celle-ci sera affiché.

1. Ouvrez le fichier KwrComment.CATPart. Ce fichier contient trois paramètres de type Longueur ainsi qu'une formule qui calcule la longueur d'une droite.
2. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier **Knowledge Advisor**.
3. Cliquez sur le paramètre Length.1 dans l'arbre de spécifications, puis sur l'icône **Commentaire & URL** ( ).
4. Dans le champ Commentaire, entrez **Length.1 est égal à Length.2: \V**. Cliquez sur **OK** lorsque vous avez terminé.
5. Cliquez avec le bouton droit de la souris sur Formula.1 dans l'arbre de spécifications, puis sélectionnez la commande **Propriétés**.
6. Dans le champ Commentaire de la boîte de dialogue Propriétés, entrez le texte suivant : Formule qui calcule la longueur de la droite.
7. Faites glisser votre curseur sur Length.1 et Formula.1 : le texte des info-bulles est modifié (voir l'illustration ci-dessous).



Utilisation de tables de paramétrage



Cliquez sur l'icône Table de paramétrage pour créer une table de paramétrage.

Présentation des tables de paramétrage
Familiarisation avec la boîte de dialogue Table de paramétrage
Création d'une table de paramétrage à partir des valeurs des paramètres du document
Création d'une table de paramétrage à partir d'un fichier existant
Ajout d'une ligne au fichier externe d'une table de paramétrage en mode interactif
Contrôle de la synchronisation des tables de paramétrage
Stockage d'une table de paramétrage dans une copie optimisée



Si vous avez déjà utilisé *CATIA* et que vous avez uniquement besoin d'un accès rapide aux informations, reportez-vous au document *CATIA* Knowledgeware Infrastructure - Tips and Techniques - Summary.

Présentation des tables de paramétrage

Une table de paramétrage :

- permet de créer et gérer des familles de composants. Ces composants peuvent être par exemple des pièces mécaniques, différant simplement dans leur valeur de paramètre.
- est un outil essentiellement destiné à faciliter la définition de pièces mécaniques. Cet outil est accessible à tous les utilisateurs de *CATIA*. Les tables de paramétrage sont particulièrement utiles dans le cadre d'une application Knowledge Advisor. Vous pouvez créer une table de paramétrage à partir d'un document *CATIA* et y exporter les données du document. Inversement, vous pouvez aussi importer dans un document CATIA les données contenues dans une table de paramétrage.
- est destinée à gérer les paramètres d'un document *CATIA* à partir de valeurs externes. Ces valeurs sont stockées sous la forme d'une table dans un fichier Microsoft ® Excel sous Windows™ ou dans un fichier texte avec tabulations. Lorsque vous utilisez une table de paramétrage, la difficulté consiste à associer comme il convient les paramètres du document avec ceux de la table. Les colonnes de la table de paramétrage peuvent ne pas correspondre à ceux du document et vous pouvez décider de n'associer qu'une partie des valeurs de la table de paramétrage au document. Pour cela, vous devez déclarer les paramètres du document à relier aux différentes colonnes de la table au moyen d'*associations*.
- devient un outil encore plus performant si vous l'utilisez avec l'application Knowledge Advisor. Cette application offre des fonctions permettant de lire les paramètres de table de paramétrage. Ces fonctions peuvent être utilisées lors de la programmation des vérification et des règles. Elles vous épargnent également les opérations d'association. Pour en savoir plus, cliquez ici.

Exemple

Les vis constituent un bon exemple de pièces mécaniques susceptibles d'être décrites par une table de paramétrage. Pour simplifier, imaginez qu'elles sont toutes décrites par quatre paramètres : la largeur de tête, la hauteur de tête, la largeur de corps et la hauteur de corps. Il est facile de regrouper les ensembles de valeurs de quatre paramètres dans une table de paramétrage. Cette dernière comporte autant de colonnes que de paramètres de vis et autant de lignes que d'ensembles de valeurs de paramètres. Dans une table de paramétrage, un ensemble de valeurs de paramètres est appelé *configuration* et est enregistré dans une ligne.

Formats de tables de paramétrage

- Format de feuille de calcul Excel (sous Windows)
- Format fichier texte avec tabulations
- Table de paramétrage CATIA

Statut d'une table de paramétrage

Formats de tables de paramétrage

Format de feuille de calcul Excel (sous Windows)

Les valeurs contenues dans les cellules de la feuille de calcul doivent être exprimées dans les unités appropriées. Dans le cas contraire, elles ne sont pas associées aux paramètres du document comme il convient.



Seules les feuilles de calcul Excel créées dans Excel 97 et les versions suivantes sont prises en charge.

Si aucune unité n'est mentionnée dans une cellule :

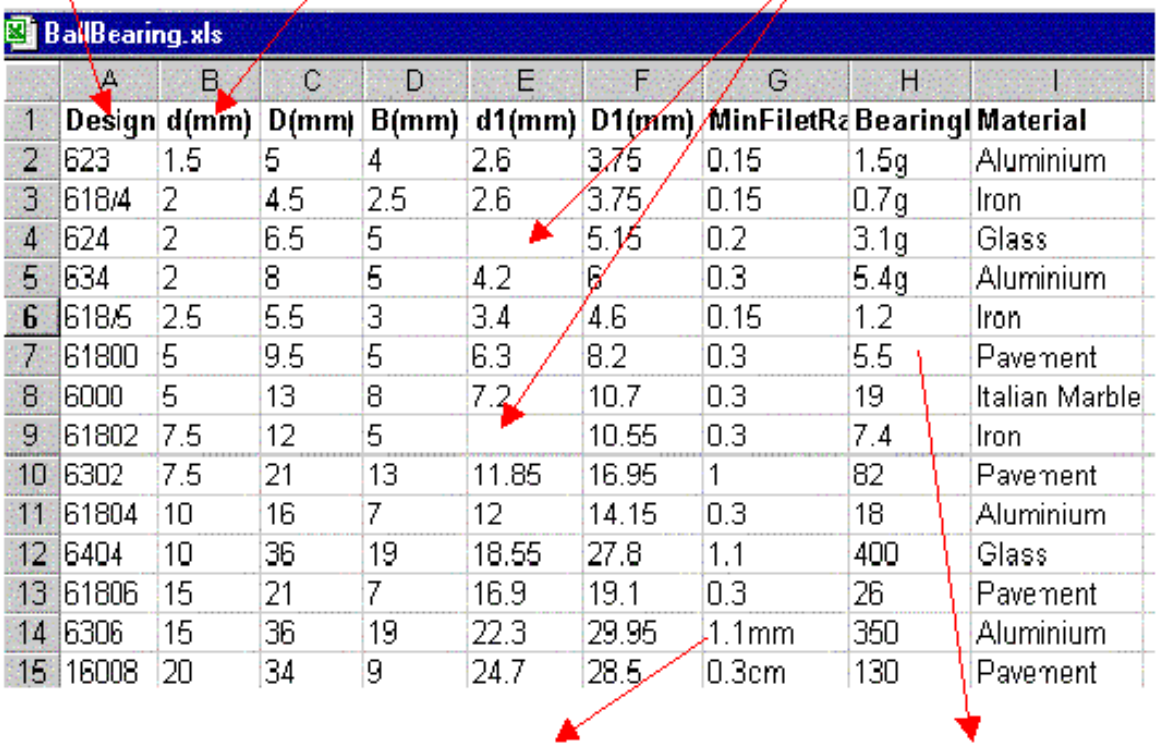
- l'unité prise en compte est celle définie dans la première ligne.
- Si aucune unité n'est spécifiée dans la première ligne, l'unité prise en compte est l'unité système SI appropriée.

Exemple de feuille de calcul Excel :

nom de la colonne

unité de la colonne

Lorsqu'une configuration contenant des cellules vides est sélectionnée, les paramètres associés à ces cellules vides ne sont pas modifiés. Cette propriété vous permet de modifier les paramètres uniquement sous certaines conditions.



BallBearing.xls

	A	B	C	D	E	F	G	H	I
1	Design	d(mm)	D(mm)	B(mm)	d1(mm)	D1(mm)	MinFiletRa	BearingI	Material
2	623	1.5	5	4	2.6	3.75	0.15	1.5g	Aluminium
3	618/4	2	4.5	2.5	2.6	3.75	0.15	0.7g	Iron
4	624	2	6.5	5		5.15	0.2	3.1g	Glass
5	634	2	8	5	4.2	6	0.3	5.4g	Aluminium
6	618/5	2.5	5.5	3	3.4	4.6	0.15	1.2	Iron
7	61800	5	9.5	5	6.3	8.2	0.3	5.5	Pavement
8	6000	5	13	8	7.2	10.7	0.3	19	Italian Marble
9	61802	7.5	12	5		10.55	0.3	7.4	Iron
10	6302	7.5	21	13	11.85	16.95	1	82	Pavement
11	61804	10	16	7	12	14.15	0.3	18	Aluminium
12	6404	10	36	19	18.55	27.8	1.1	400	Glass
13	61806	15	21	7	16.9	19.1	0.3	26	Pavement
14	6306	15	36	19	22.3	29.95	1.1mm	350	Aluminium
15	16008	20	34	9	24.7	28.5	0.3cm	130	Pavement

Dans une colonne donnée, vous pouvez changer les unités.

Les unités peuvent être mentionnées dans des cellules. Aucune unité = SI

Il est vivement recommandé de choisir le format Général et non le format Cellules dans Excel.

Format fichier texte avec tabulations

Voici un exemple de fichier en format texte avec tabulations. Cette table de paramétrage peut être créée avec n'importe quel éditeur de texte. Utilisez la touche de tabulation pour passer d'une colonne à une autre. Les règles appliquées aux unités sont les mêmes que celles des feuilles de calcul Excel.

BallBearing0 - Notepad

File Edit Search Help

Designation	d(mm)	D(mm)	B(mm)	d1(mm)	D1(mm)	MinFiletRadius(mm)
623	1.5	5	4	2.6	3.75	0.15	1.5
618/4	2	4.5	2.5	2.6	3.75	0.15	0.7
624	2	6.5	5	3.35	5.15	0.2	3.1
634	2	8	5	4.2	6	0.3	5.4
618/5	2.5	5.5	3	3.4	4.6	0.15	1.2
61800	5	9.5	5	6.3	8.2	0.3	5.5
6000	5	13	8	7.2	10.7	0.3	19
61802	7.5	12	5	8.95	10.55	0.3	7.4
6302	7.5	21	13	11.85	16.95	1	82
61804	10	16	7	12	14.15	0.3	18
6404	10	36	19	18.55	27.8	1.1	400
61806	15	21	7	16.9	19.1	0.3	26
6306	15	36	19	22.3	29.95	1.1	350
16008	20	34	9	24.7	28.5	0.3	130
6308	20	45	23	28.05	37.35	1.5	63000

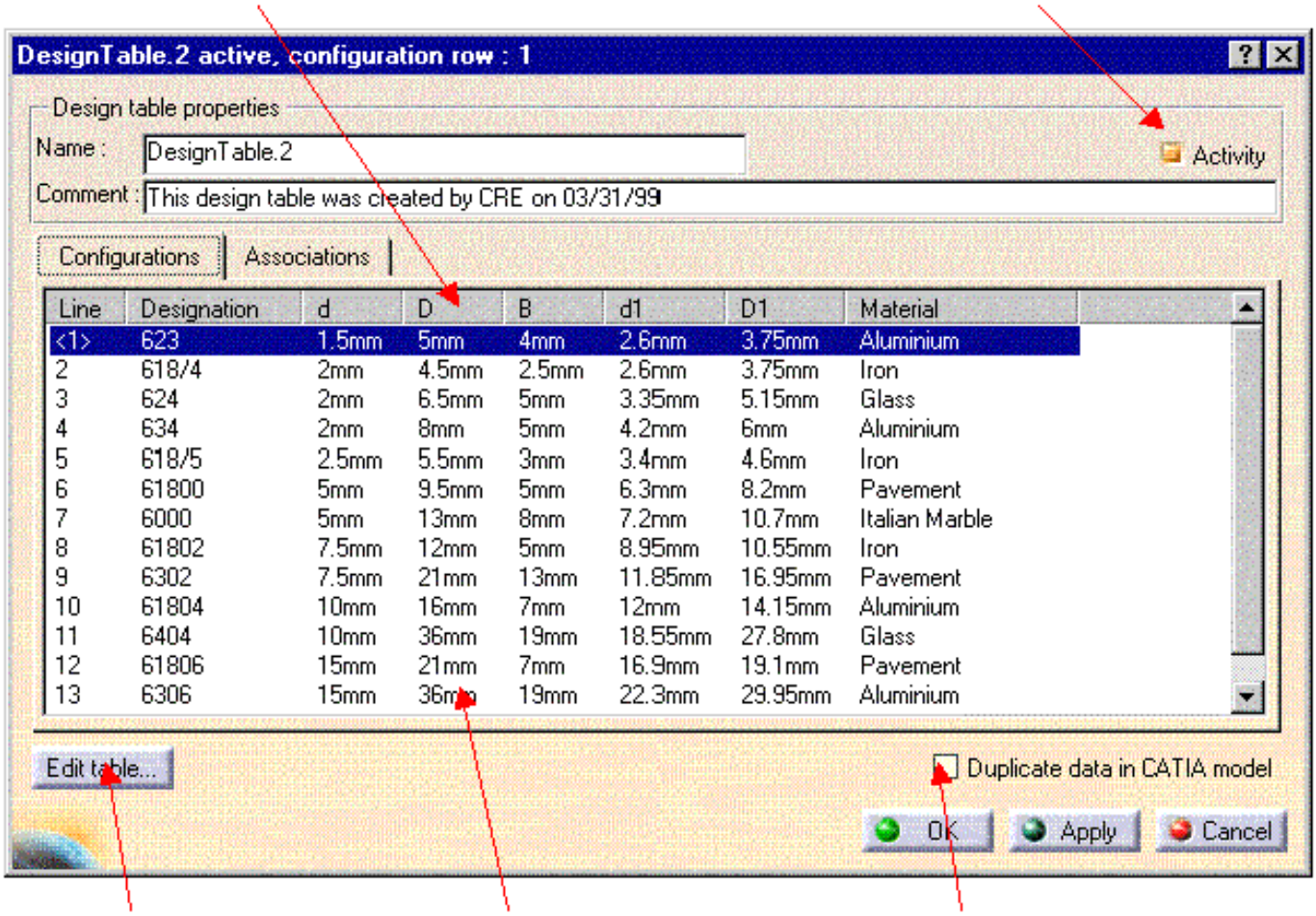
Sous UNIX, il est possible de changer l'éditeur de table de paramétrage par défaut. Pour ce faire, entrez : **export CATTextEditorDT=... (indiquez le chemin de l'éditeur.)**

Table de paramétrage CATIA

Une fois lue et traitée par *CATIA*, la table de paramétrage ressemble à ceci :

Aucune unité dans une colonne

Cochez la case pour modifier l'activité



Line	Designation	d	D	B	d1	D1	Material
<1>	623	1.5mm	5mm	4mm	2.6mm	3.75mm	Aluminium
2	618/4	2mm	4.5mm	2.5mm	2.6mm	3.75mm	Iron
3	624	2mm	6.5mm	5mm	3.35mm	5.15mm	Glass
4	634	2mm	8mm	5mm	4.2mm	6mm	Aluminium
5	618/5	2.5mm	5.5mm	3mm	3.4mm	4.6mm	Iron
6	61800	5mm	9.5mm	5mm	6.3mm	8.2mm	Pavement
7	6000	5mm	13mm	8mm	7.2mm	10.7mm	Italian Marble
8	61802	7.5mm	12mm	5mm	8.95mm	10.55mm	Iron
9	6302	7.5mm	21mm	13mm	11.85mm	16.95mm	Pavement
10	61804	10mm	16mm	7mm	12mm	14.15mm	Aluminium
11	6404	10mm	36mm	19mm	18.55mm	27.8mm	Glass
12	61806	15mm	21mm	7mm	16.9mm	19.1mm	Pavement
13	6306	15mm	36mm	19mm	22.3mm	29.95mm	Aluminium

Affiche les données brutes de la table de paramétrage.

Valeurs avec unités

Duplique les données externes de la table de paramétrage dans le document *CATIA*. Cochez cette case chaque fois que vous souhaitez avoir de nouveau accès à votre table de paramétrage sur une autre plate-forme.

Statut de tables de paramétrage :

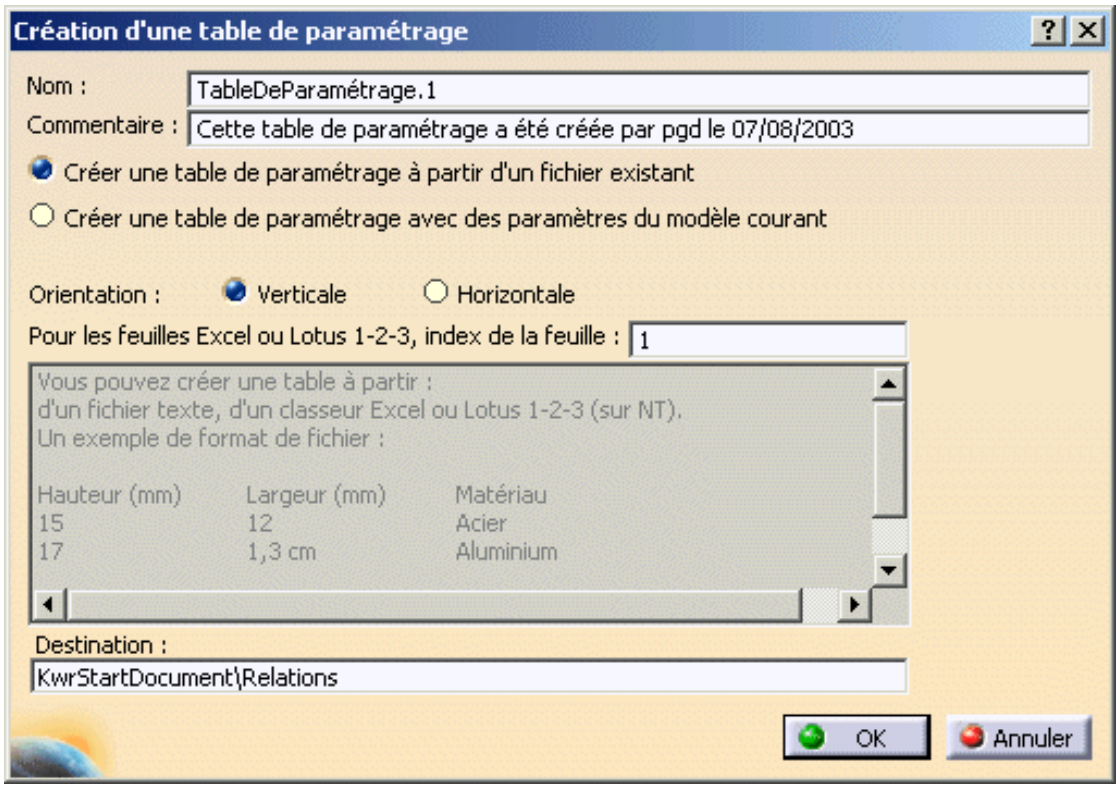
Plusieurs statuts sont possibles pour une table de paramétrage :

- Synchronisé : la table de paramétrage est copiée dans le modèle (option "Dupliquer les données dans le modèle CATIA" activée) et les données enregistrées dans le modèle sont synchronisées avec le contenu du fichier.
- Non synchronisé : la table de paramétrage est copiée dans le modèle (option "Dupliquer les données dans le modèle CATIA" activée) et les données enregistrées dans le modèle sont synchronisées avec le contenu du fichier. (Le fichier .xls ou .txt a été modifié à l'extérieur de CATIA.)
- Inactif : la table de paramétrage est désactivée. Les paramètres associés ne sont plus gérés par la table de paramétrage.
- Irrésolu : la table de paramétrage est fonctionnellement KO car elle ne peut pas être utilisée pour évaluer les paramètres associés (l'accès aux données est impossible car le fichier est perdu ou 2 colonnes portent le même nom...). .
- Light Broken : le filtre n'est pas compatible avec la configuration en cours, ou la valeur de la configuration en cours ne correspond pas à l'une des lignes du fichier.

Familiarisation avec la boîte de dialogue Table de paramétrage

Voici la séquence de boîte de dialogue que vous obtenez à l'écran lorsque vous cliquez sur l'icône  dans la barre d'outils standard.

Création d'une table de paramétrage



Option ***"Créer une table de paramétrage à partir d'un fichier existant"***

Sélectionnez cette option chaque fois que vous voulez créer une table de paramétrage à partir des valeurs d'un fichier externe. Dans ce cas, la table créée contient :

- soit uniquement les colonnes dont le nom correspond à un nom de paramètre du document. Si le fichier externe contient une colonne "Longueur", mais qu'il n'y a pas de paramètre"Longueur" défini dans le document, la colonne "Longueur" n'apparaîtra pas dans la table de paramétrage créée. C'est ce qu'on appelle le processus d'association "automatique".

- soit, uniquement les colonnes qui ont été associées une par une à un paramètre du document. Si le fichier externe contient une colonne "Longueur", mais qu'il n'y a aucun paramètre de ce nom dans le document, vous pouvez choisir d'associer cette colonne à un paramètre de même type (rayon d'esquisse, par exemple).

Option ***"Créer une table de paramétrage avec des paramètres du modèle courant"***

Sélectionnez cette option chaque fois que vous voulez créer une table de paramétrage à partir d'un sous-ensemble de paramètres du document. Il vous suffit alors de sélectionner, parmi les paramètres du document, ceux que vous voulez inclure comme colonnes dans la table de paramétrage. Dans ce cas, la table créée ne contient qu'une seule ligne.

Options ***Orientation***

Ces options permettent de choisir l'orientation de la table de paramétrage. Une orientation verticale est recommandée lorsque la table de paramétrage contient un grand nombre de paramètres.

Index de la ***feuille***

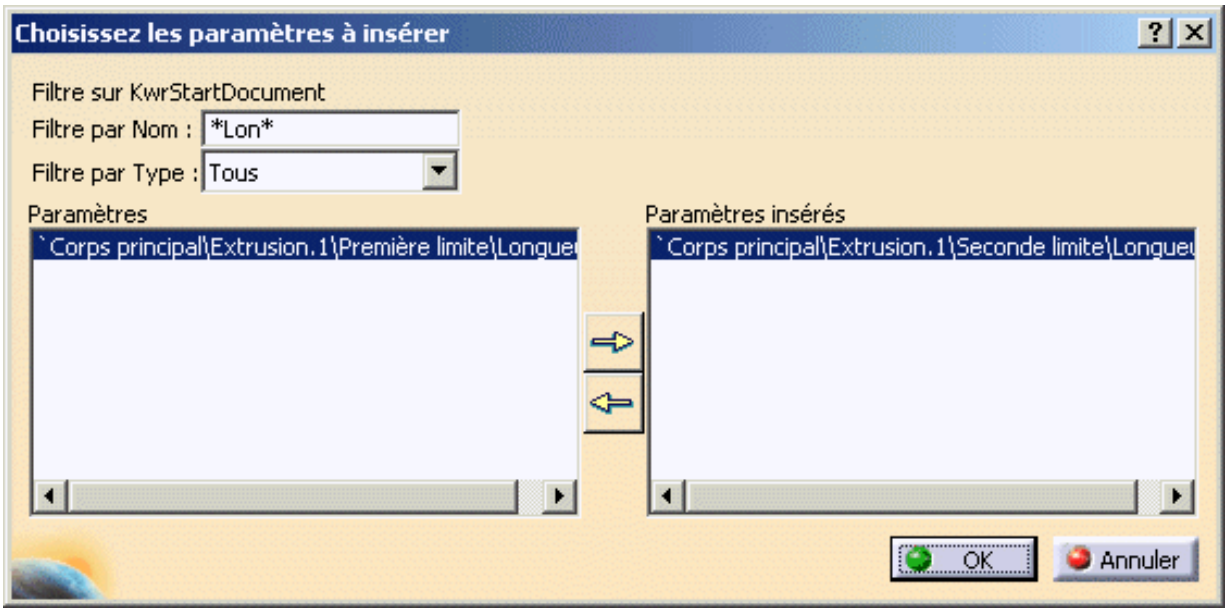
A partir de la version 5 édition 7, vous pouvez indiquer un numéro de feuille Excel ou Lotus.

Champ ***Destination***

Toutes les relations de Knowledgeware telles que les tables de paramétrage, les règles, les vérifications ou les formules sont créées par défaut sous le noeud Relations. La création d'une relation sous un composant donné peut vous aider à organiser votre document. Pour définir une destination, sélectionnez la destination par défaut dans le champ Destination, puis cliquez sur le composant qui servira de nouvelle destination dans l'arbre de spécifications ou dans la zone géométrique.

Sélection des paramètres à insérer

Cette boîte de dialogue apparaît lorsque vous cochez la case "Créer une table de paramétrage avec des paramètres du modèle courant".

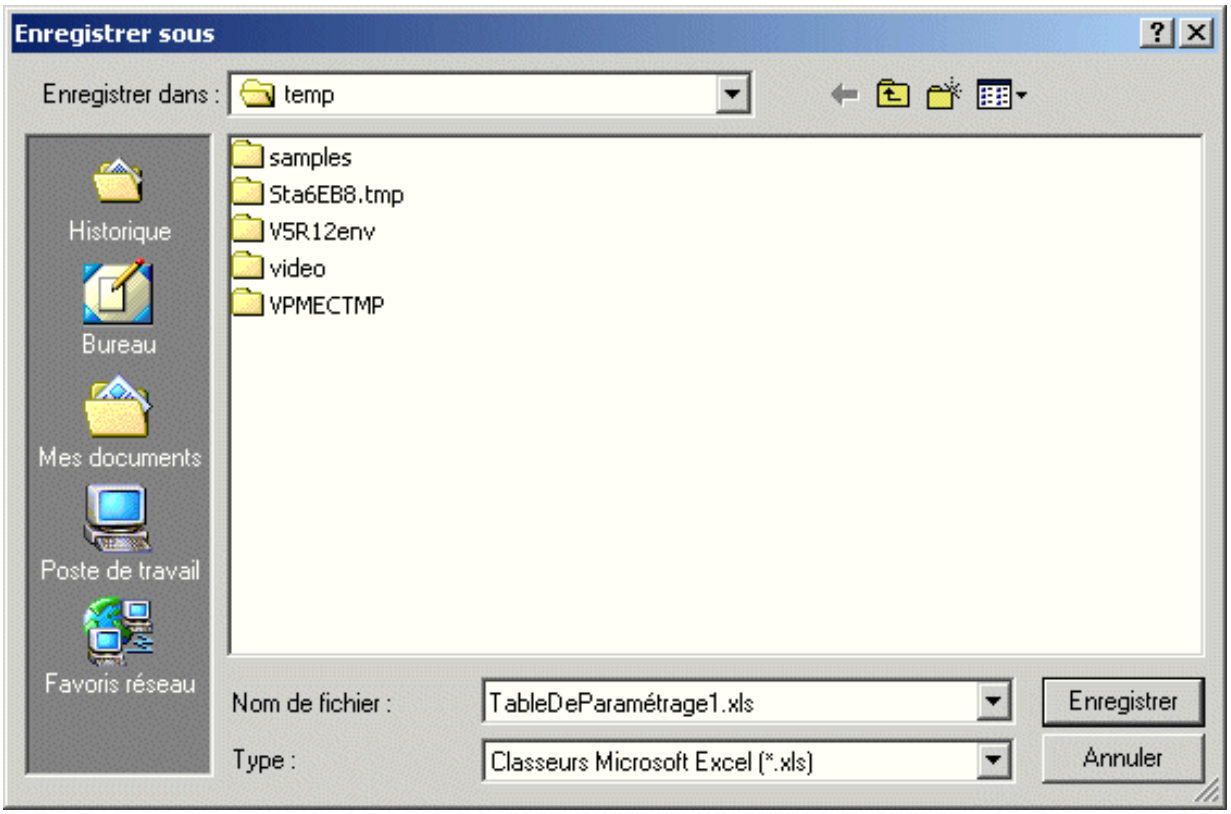


Pour limiter la liste des paramètres à insérer, deux méthodes sont possibles. Vous pouvez utiliser :

1. le champ *Filtre par nom*
   Utilisez le caractère * pour spécifier une chaîne quelconque incluse dans un nom de paramètre. Par exemple, si vous indiquez *Lon*, la liste "Paramètres" de la boîte de dialogue contiendra tous les paramètres dont le nom contient la chaîne Lon.
2. et le champ *Filtre par type* .

Lorsque vous cliquez sur OK dans la boîte de dialogue ci-dessus, la fenêtre "Choisissez le chemin et le nom du fichier de la table" s'affiche.

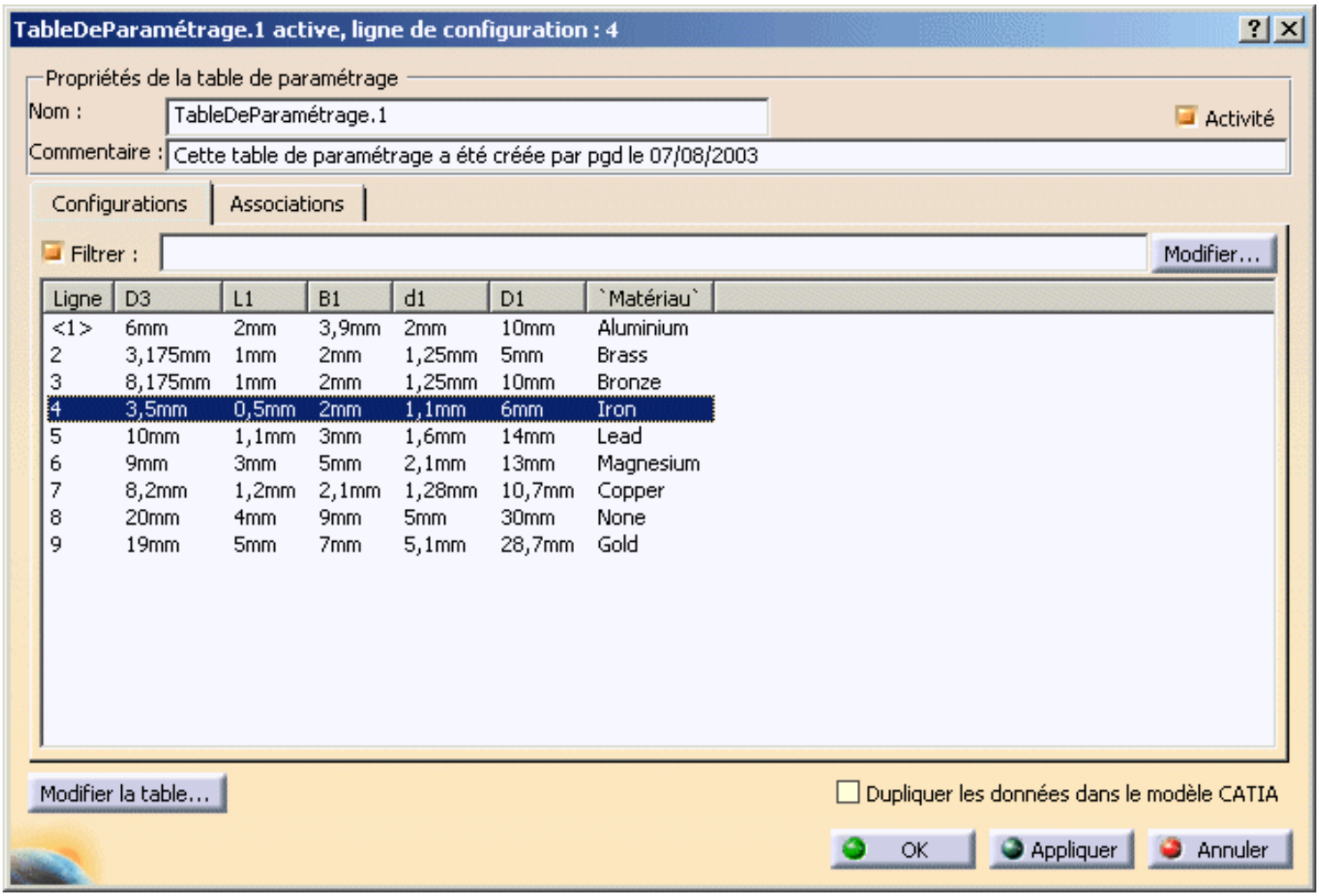
Sélection du fichier à créer



Utilisez cette boîte de dialogue pour indiquer le fichier .xls (Windows) ou .txt à créer. Spécifiez l'extension .xls lorsque vous complétez le champ "Nom du fichier". Cliquez ensuite sur Ouvrir pour afficher la boîte de dialogue Table de paramétrage.

Boîte de dialogue Table de paramétrage

Onglet "Configurations"

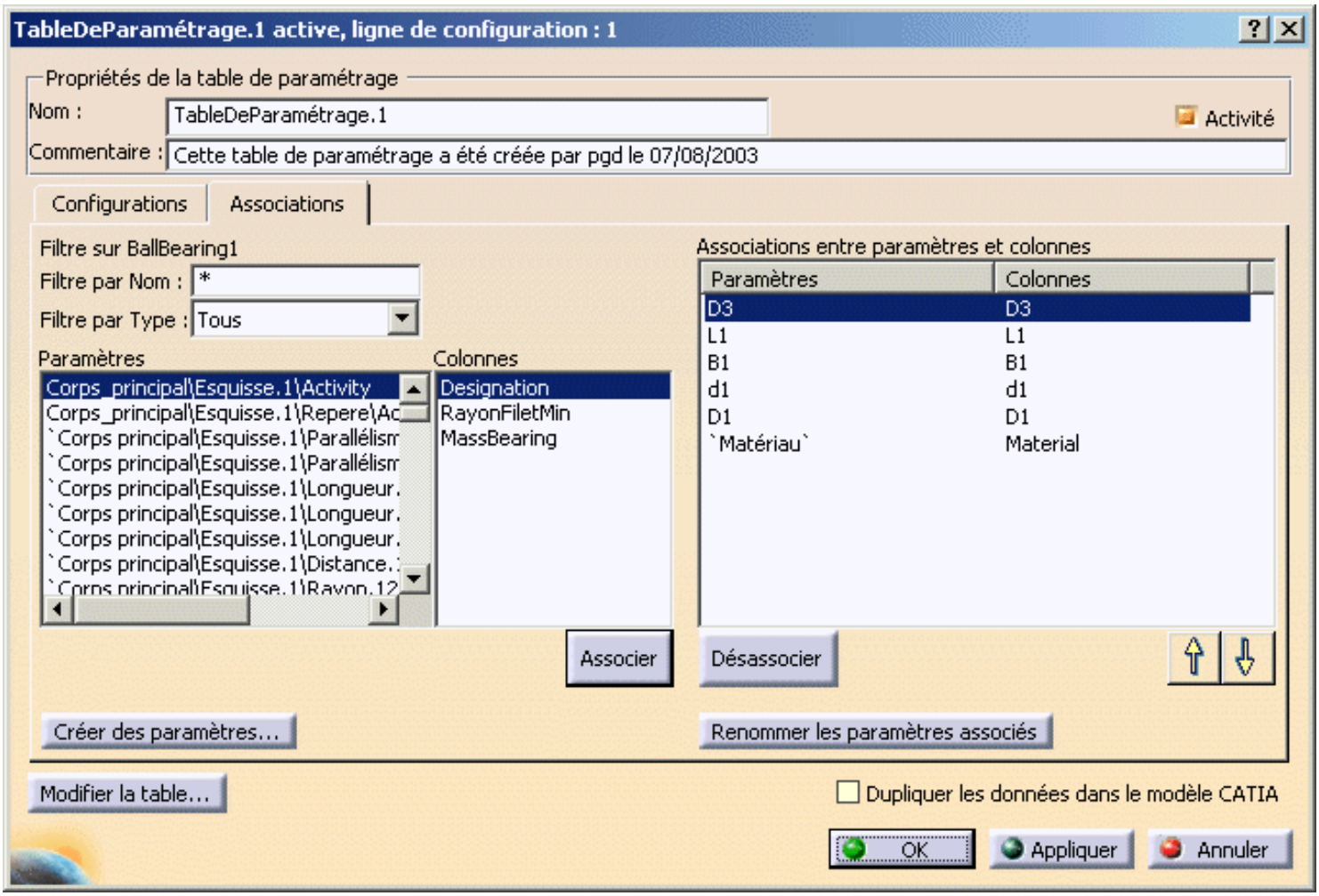


La configuration en cours, tout comme son numéro (< *numéro de configuration*>) sont mis en évidence. Pour changer de configuration, il suffit de cliquer sur la nouvelle configuration dans la table de paramétrage.
Lorsqu'une table de paramétrage est créée à partir des valeurs des paramètres en cours, elle ne contient qu'une seule ligne.

- ***Filtre***
  Le filtre permet de rechercher facilement une configuration répondant à certains critères. Cliquez sur le bouton "Modifier... " pour afficher l'"Editeur de requêtes de table de paramétrage". Pour plus d'informations sur l'utilisation de la syntaxe fournie par le dictionnaire, reportez-vous à la section Utilisation du dictionnaire.
  Dans une requête, vous pouvez spécifier des conditions faisant référence aussi bien à des paramètres de la table de paramétrage qu'à des paramètres externes à celle-ci.
- ***Option "Activité"***
  Lorsque vous créez une table de paramétrage, celle-ci est par défaut active. L'option Activité vous permet de la désactiver.
- ***Option "Modifier la table..."***
  Activez cette option pour afficher la table à créer. En fonction de l'extension que vous avez choisie ((.xls ou .txt), vous lancerez l'application Microsoft Excel ou votre éditeur de texte par défaut.
- **Option "Dupliquer les données dans le modèle CATIA"**
  Cochez cette case chaque fois que vous envisagez de réutiliser votre document sous un système d'exploitation différent de celui utilisé pour créer la table de paramétrage. Ainsi, la table de paramétrage sera dupliquée dans votre document.

Onglet "Associations"

Cet onglet vous permet d'associer les paramètres d'un document aux colonnes de la table de paramétrage externe. La partie gauche de la boîte de dialogue sert à associer les paramètres aux colonnes de la table de paramétrage ; la partie droite affiche la liste des paramètres associés.



- ***Option "Créer des paramètres..."***
  Lorsqu'une table de paramétrage fait référence à un paramètre qui n'a pas été créé dans le document, vous pouvez utiliser cette option pour créer un paramètre dans le document et l'associer à la colonne correspondante de la table de paramétrage.

- ***Option "Renommer les paramètres associés"***

  Si un paramètre ne porte pas le même nom que la colonne à laquelle il est associé, vous pouvez renommer ce paramètre, de manière à ce que les deux noms correspondent. Lorsque vous activez l'option "Renommer les paramètres associés", une boîte de dialogue apparaît, vous demandant si vous voulez renommer tous les paramètres ou seulement certains d'entre eux.

Création d'une table de paramétrage à partir des valeurs des paramètres en cours

Une table de paramétrage se crée à partir des paramètres en cours d'un document ou à partir de données externes. Que des données externes existent ou non, vous devez **créer** la table de paramétrage dans *CATIA*. Deux méthodes sont possibles pour créer une table de paramétrage :

- A partir des valeurs de paramètre en cours
- A partir d'un fichier existant

Le scénario décrit ci-après explique comment procéder dans le premier cas. Le processus de création d'une table de paramétrage se décompose en plusieurs étapes :

a. Créez une table à partir des paramètres du document.

b. Sélectionnez les paramètres à ajouter à la table de paramétrage.

c. Spécifiez un fichier devant contenir la table de paramétrage générée.

d. Editez la table de paramétrage *CATIA* générée.

e. Appliquez la table de paramétrage au document.

Pour plus d'informations sur l'utilisation des différentes boîtes de dialogue associées à la table de paramétrage, reportez-vous à la section Boîte de dialogue Table de paramétrage.

1. Ouvrez le document KwrStartDocument.CATPart.
2. Cliquez sur l'icône  Table de paramétrage dans la barre d'outils standard. La boîte de dialogue Création d'une table de paramétrage s'affiche. Pour plus d'informations, reportez-vous à la section Boîte de dialogue Table de paramétrage .
3. Si besoin est, remplacez le nom par défaut et le commentaire de la table de paramétrage.
4. Activez l'option **Créer une table de paramétrage avec des paramètres**

du modèle courant.

5. Cliquez sur OK. La boîte de dialogue **Choisissez les paramètres à insérer** s'affiche.
6. Dans la liste Paramètres publiables, sélectionnez les composants PartBody\Pad.1\FirstLimit\Length et PartBody\Pad.1\SecondLimit\Length. Cliquez ensuite sur la flèche droite pour ajouter les deux composants à la liste Paramètres insérés.
7. Cliquez sur **OK**. Une boîte de sélection de fichier s'affiche.
8. Indiquez le nom de chemin de la table de paramétrage à créer. Cliquez sur **OK** dans la boîte de dialogue de sélection de fichier.
   Le composant de table de paramétrage est ajouté dans l'arbre de spécifications et une boîte de dialogue affiche la nouvelle table de paramétrage. Cette table de paramétrage contient une seule configuration. Par défaut, cette dernière est active.

   Si le fichier spécifié existe déjà, la boîte de dialogue Création d'une table de paramétrage s'affiche à nouveau, ainsi qu'un message vous demandant si vous souhaitez écraser le fichier existant.

9. Cliquez sur **Modifier la table...** pour démarrer une application Excel (sous Windows) ou ouvrez l'éditeur de texte sous Unix.
   Remplacez la valeur du paramètre PartBody\Pad.1\FirstLimit\Length par 80 mm.
10. Enregistrez votre fichier Excel ou .txt, puis fermez l'application. Certains messages d'information s'affichent dans une boîte de dialogue pour notifier les événements éventuels liés à la table de paramétrage. Cliquez sur Fermer.
11. Cliquez sur **Appliquer** dans la table de paramétrage *CATIA* : le document est mis à jour, de même que la table de paramétrage *CATIA*. Cliquez sur OK pour quitter la boîte de dialogue et ajouter la table de paramétrage au document.

Création d'une table de paramétrage à partir d'un fichier existant

Une table de paramétrage se crée à partir des paramètres d'un document ou à partir de données externes. Que des données externes existent ou non, vous devez **créer** la table de paramétrage dans *CATIA*. Deux méthodes sont possibles pour créer une table de paramétrage :

- A l'aide des valeurs de paramètre en cours
- A l'aide d'un fichier existant

Le scénario décrit ci-après explique comment procéder dans le second cas. Voici les étapes principales à suivre :

a. Sélectionnez le fichier existant contenant les données brutes.
b. Créez les associations entre les paramètres de document et les colonnes de table externes. Vous pouvez choisir de créer ces associations automatiquement.
c. Editez la table de paramétrage *CATIA* générée.
d. Sélectionnez une configuration dans la table de paramétrage générée. Vous pouvez modifier la configuration par défaut proposée par *CATIA*.
e. Appliquez la table de paramétrage au document.

Pour plus d'informations sur l'utilisation des différentes boîtes de dialogue associées à la table de paramétrage, reportez-vous à la section Boîte de dialogue Table de paramétrage.

Il est à présent possible de sélectionner un fichier externe de table de paramétrage à partir d'un VPDM (ENOVIA LCA, ...). Pour ce faire, vérifiez que vous avez activé l'environnement voulu dans le champ Environnements de document (**Outils -> Options -> Général -> Document**.) Vos documents sont accessibles via la fonction Localisation des documents.

1. Ouvrez le document KwrStartDocument.CATPart.

2. Cliquez sur l'icône Table de paramétrage () dans la barre d'outils standard.

   La boîte de dialogue **Création d'une table de paramétrage** s'affiche.

   Entrez un nom (Table_paramétrage1 par exemple) et un commentaire.

3. Activez l'option **Créer une table de paramétrage à partir d'un fichier existant**. Cliquez sur **OK**.

4. Sélectionnez le fichier KwrBallBearing.xls et cliquez sur Ouvrir. Une boîte de dialogue vous demande si vous souhaitez effectuer des associations automatiques entre les colonnes de la table de paramétrage et les paramètres du document du même nom.

5. Cliquez sur **Oui**. Le paramètre **Matériau** est le seul qui est commun aux paramètres de document et à la table de paramétrage externe. Une table de paramétrage à plusieurs lignes est créée. Les symboles '<' et '>' indiquent la configuration en cours.

6. Sélectionnez la configuration à appliquer au document (ligne 4 par exemple). Cliquez sur **Appliquer**.
   La valeur du paramètre Fer s'affiche dans l'arbre de spécifications.

7. Cliquez sur **OK** pour achever la création de la table de paramétrage.





Le scénario ci-après illustre comment créer une table de paramétrage en associant un par un les paramètres du document aux colonnes du fichier d'entrée.



1. Ouvrez le document KwrStartDocument.CATPart.

2. Cliquez sur l'icône  Table de paramétrage dans la barre d'outils standard.
   La boîte de dialogue "Création d'une table de paramétrage" s'affiche. Entrez un nom (Table_paramétrage2 par exemple) et un commentaire.

3. Activez l'option Créer une table de paramétrage à partir d'un fichier existant. Cliquez sur **OK**. Un écran de sélection de fichier s'affiche.

4. Sélectionnez le fichier KwrBallBearing.xls. Cliquez sur **Ouvrir**. La boîte de dialogue Associations automatiques s'affiche.

5. Cliquez sur **Non**. La boîte de dialogue de table de paramétrage vous informe qu'il n'y a aucune association entre les paramètres et les colonnes.

Vous devez, à présent, associer un à un les paramètres du document avec les colonnes de la table de paramétrage.

6. Cliquez sur l'option **Associations**. La boîte de dialogue de table de paramétrage affiche désormais côte à côte la liste des paramètres du document et les colonnes du fichier d'entrée.

7. Dans la liste Paramètres, sélectionnez l'élément Corps_Principal\Trou.1\Diamètre. Dans la liste Colonnes, sélectionnez le paramètre d1. Cliquez ensuite sur **Associer**. Un couple de paramètres apparaît à présent dans la liste Associations entre paramètres et colonnes.

8. Répétez la même opération pour le paramètre Matériau.

La sélection d'un paramètre ou d'une association dans la liste met en évidence les valeurs correspondantes dans la zone géométrique.



La liste des paramètres peut être filtrée :

- Cliquez sur un composant (dans l'arbre de spécifications ou dans la zone géométrique). Toutes les valeurs de paramètre du composant sélectionné (et les objets enfants associés) sont mises en évidence dans la zone géométrique. La liste des paramètres affiche uniquement les paramètres des composants sélectionnés (et leurs enfants).

- Entrez une chaîne dans la zone Filtre par nom. Par exemple, pour afficher tous les paramètres de longueur, tapez *ongueur*.

- Indiquez un type dans la zone Filtre par type.

Le bouton **Créer des paramètres...** permet de créer automatiquement des paramètres et des associations pour les composants de la liste Colonnes. Le bouton Renommer les paramètres associés remplace le nom du paramètre par celui de la colonne.

9. Cliquez sur **OK** pour achever la création de la table Table_paramétrage2.

   Le composant Table_paramétrage2 est ajouté comme relation dans l'arbre de spécifications. Cliquez deux fois sur Table_paramétrage2 dans l'arbre de spécifications pour modifier la table. Par défaut, la configuration 1 est appliquée au document. Un nouveau matériau (Aluminum) est appliqué au document et le diamètre du trou est modifié. Vous pouvez sélectionner une autre configuration et l'appliquer à votre document.



Ajout interactif d'une ligne au fichier externe de la table de paramétrage

La tâche décrite ci-après explique comment ajouter une ligne à un fichier externe de table de paramétrage. Le scénario se compose des étapes suivantes :

- L'utilisateur ouvre le fichier CATPart et insère la table de paramétrage.
- L'utilisateur désactive la table de paramétrage et crée une nouvelle configuration.
- L'utilisateur ajoute la configuration au fichier externe de la table de paramétrage.
- L'utilisateur active la table de paramétrage et met en oeuvre la nouvelle configuration.

Cette nouvelle fonction permet à l'utilisateur d'associer un menu contextuel à la table de paramétrage (dans l'arborescence) qui apparaît uniquement dans les conditions suivantes :

- Si la table de paramétrage est désactivée.
- Si le fichier externe de la table de paramétrage existe et est accessible en lecture et en écriture.
- Si au moins un paramètre est associé.

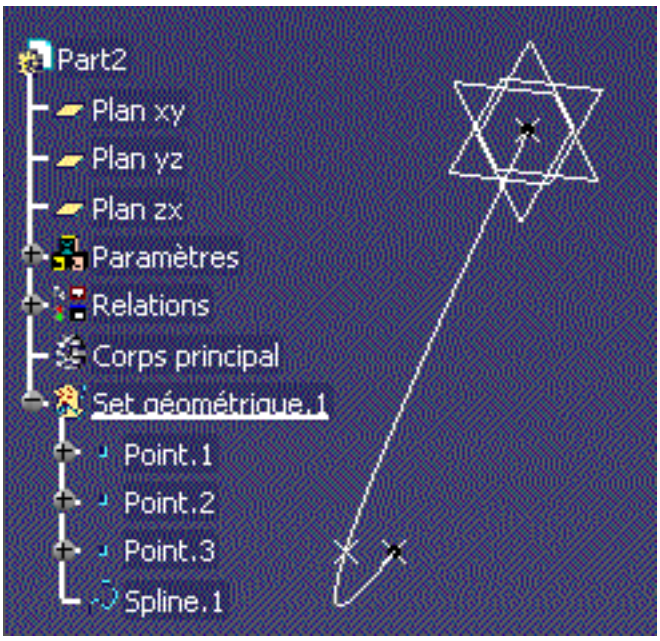
Cette commande a pour effet d'ajouter une ligne à la fin du fichier de la table de paramétrage contenant les valeurs des paramètres associés. Une cellule vide est ajoutée pour les colonnes non associées.

Pour effectuer ce scénario, l'utilisateur a besoin des fichiers suivants
KwrAddARow.CATPart
KwrAddARow.xls

Notez que cette tâche ne peut être exécutée que dans un environnement anglais.

Avant d'effectuer ce scénario, vérifiez que les options **Avec valeur** et **Avec formule** sont actives dans l'onglet **Outils -> Options -> Général -> Paramètres et mesure -> Knowledge**.

1. Ouvrez le fichier KwrAddARow.CATPart. L'image suivante s'affiche :



2. Cliquez sur l'icône Table de paramétrage( ).
3. Activez l'option **Créer une table de paramétrage à partir d'un fichier existant** et cliquez sur **OK**.
4. Dans la fenêtre **Sélection de fichier** qui apparaît, sélectionnez le fichier KwrAddARow.xls et cliquez sur**Ouvrir**.
5. Cliquez sur **Oui** dans la fenêtre Associations automatiques : la table de paramétrage s'ouvre. Cliquez sur **OK** pour la fermer.
6. Cliquez sur l'icône **Mise à jour des mesures** pour procéder à la mise à jour de Formule.1.
7. Sous le noeud Tables de paramétrage, cliquez deux fois sur **Configuration=1**. La boîte de dialogue **Editer paramètre** apparaît.
8. Cliquez sur l'icône Table de paramétrage dans la boîte de dialogue **Editer paramètre** : la fenêtre Table de paramétrage s'affiche.
9. Dans la boîte de dialogue, sélectionnez la deuxième configuration (ligne 2), cliquez sur **Appliquer**, puis deux fois sur **OK**.

10. Cliquez avec le bouton droit de la souris sur Formule.1 dans l'arbre de spécifications et sélectionnez l'option **Mise à jour locale**.

11. Dans l'arbre de spécifications, cliquez à l'aide du bouton droit de la souris sur Table_paramétrage.1, puis sélectionnez la commande **DesignTable.1 objet -> Désactiver**. La table de paramétrage est désactivée.

12. Modifiez la spline en procédant comme suit :

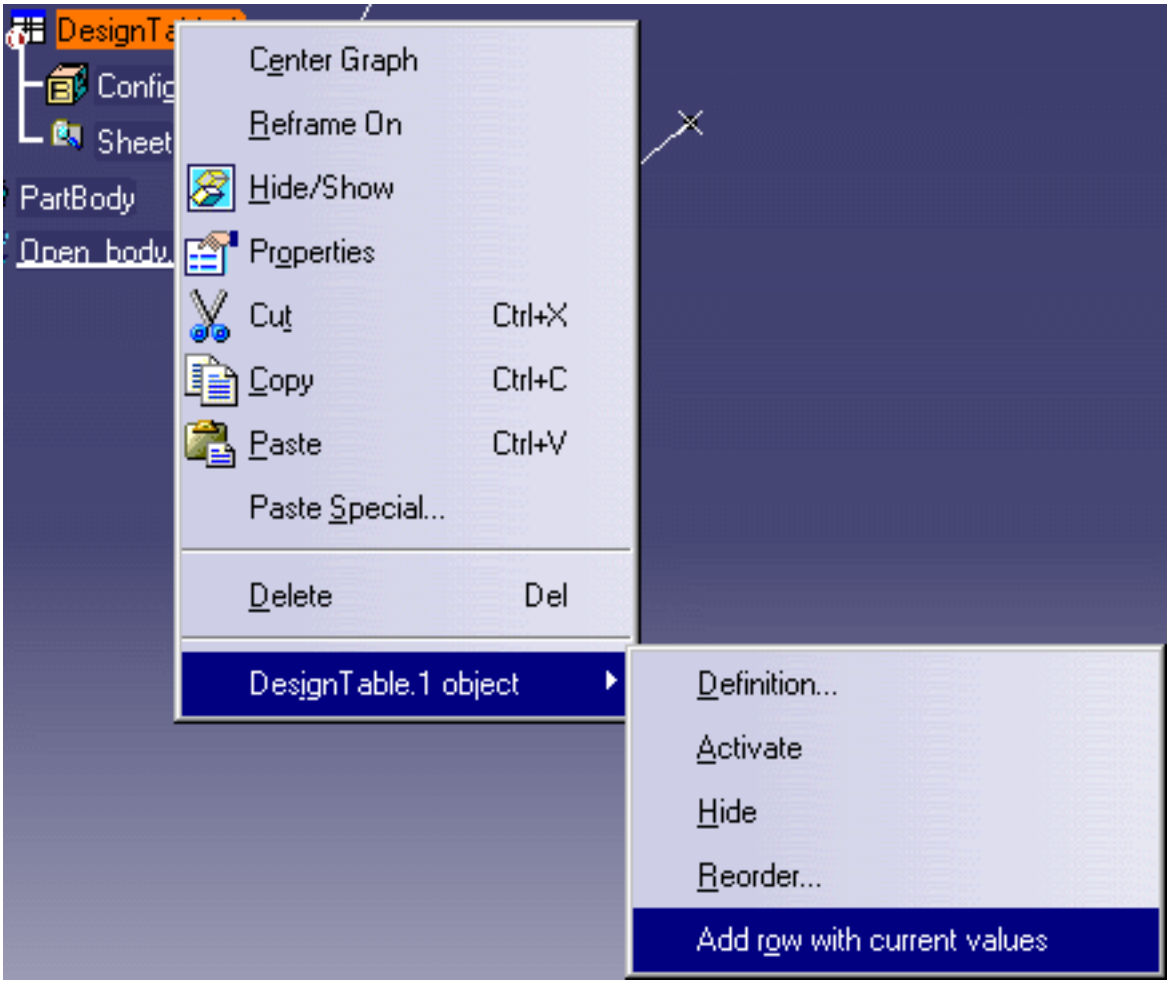
- Cliquez deux fois sur Point.1 dans l'arbre de spécifications ou dans la géométrie. Entrez les coordonnées ci-dessous dans la boîte de dialogue Définition de point.

- Modifiez les coordonnées de Point.2 et de Point.3 (voir le tableau en regard)

	Point 1	**Point 2**	**Point 3**
X	0	100	50
Y	0	100	50
Z	0	-100	-226

- Lorsque vous avez terminé, cliquez sur **OK**.

13. Ajoutez la nouvelle configuration dans la table de paramétrage. Pour ce faire, cliquez avec le bouton droit sur Table_paramétrage.1 dans l'arbre de spécifications et choisissez l'option **DesignTable.1 objet -> Ajouter une ligne avec les valeurs courantes**.



14. Cliquez à l'aide du bouton droit de la souris sur Table_paramétrage.1, puis sélectionnez la commande **DesignTable.1 objet -> Activer**.

15. Cliquez deux fois sur **Configuration=1** sous Table_paramétrage.1, puis cliquez sur l'icône Table de paramétrage( ).

16. Dans la fenêtre Table_paramétrage.1, sélectionnez la configuration que vous venez d'ajouter, cliquez sur **Appliquer**, puis deux fois sur **OK**. La spline est mise à jour en conséquence.

Contrôle de la synchronisation des tables de paramétrage

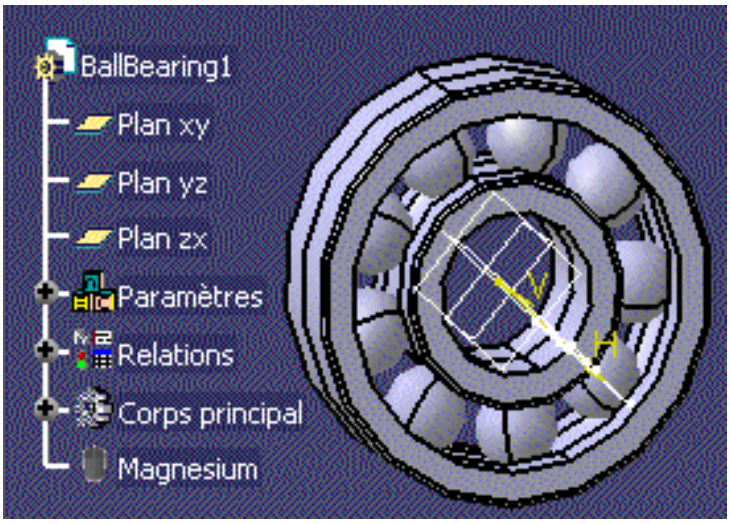
Dans cette tâche, vous trouverez de brefs exemples d'utilisation des tables de paramétrage dans les modes suivants :

- Synchronisation automatique lors du chargement
- Synchronisation interactive lors du chargement
- Synchronisation manuelle

Synchronisation automatique lors du chargement

Lors du chargement d'un modèle contenant des tables de paramétrage définies par l'utilisateur, si les fichiers correspondants ont été modifiés et que le modèle contient les données des fichiers externes, la table de paramétrage est synchronisée automatiquement lorsque cette option est sélectionnée.

1. Ouvrez le fichier KwrBallBearing1.CATPart. L'image suivante s'affiche :



2. Cliquez sur l'icône **Table de paramétrage** ( ). La boîte de dialogue Création d'une table de paramétrage apparaît.

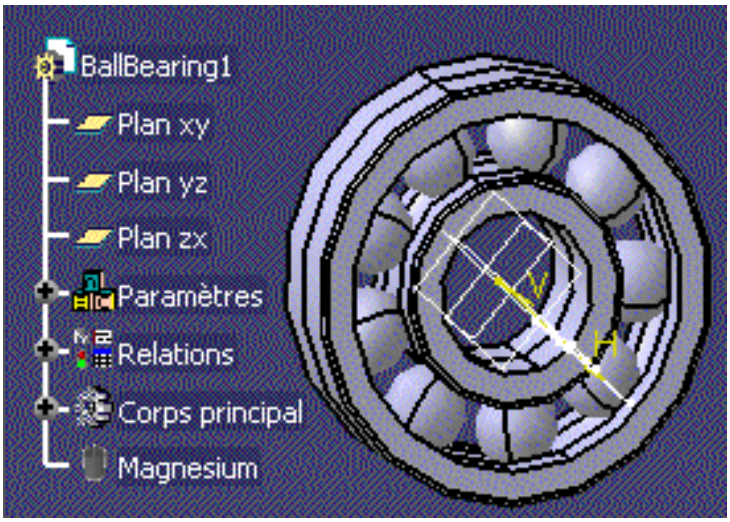
3. Activez l'option **Créer une table de paramétrage à partir d'un fichier existant** et cliquez sur **OK**. La boîte de dialogue Sélection de fichiers s'ouvre.

4. Sélectionnez le fichier KwrBearingDesignTable.xls et cliquez sur **Ouvrir**.
   Cliquez sur **Oui** lorsque vous êtes invité à préciser si vous voulez associer les colonnes des tables aux paramètres.
5. Cliquez sur **OK** pour appliquer la configuration par défaut.
6. Enregistrez le fichier, puis fermez-le.
7. Ouvrez le fichier KwrBearingDesignTable.xls. Modifiez le matériau de la ligne 2 en Or. Enregistrez le fichier, puis fermez-le.
8. Revenez dans Catia. Ouvrez la pièce : elle est mise à jour en fonction des modifications apportées.

Synchronisation interactive lors du chargement

Lors du chargement d'un modèle contenant des tables de paramétrage définies par l'utilisateur dont le fichier source a été supprimé, cette option permet à l'utilisateur de sélectionner un nouveau fichier source ou d'enregistrer les données de la table de paramétrage dans un nouveau fichier.

1. Dans le menu **Outils -> Options...**, sélectionnez **Général -> Paramètres et mesure** et activez l'option **Synchronisation interactive au chargement** dans l'onglet Knowledge.
2. Ouvrez le fichier KwrBallBearing1.CATPart. L'image suivante s'affiche :



3. Cliquez sur l'icône Table de paramétrage( ). La boîte de dialogue Création

d'une table de paramétrage apparaît.

4. Activez l'option **Créer une table de paramétrage à partir d'un fichier existant** et cliquez sur **OK**. La boîte de dialogue Sélection de fichiers s'ouvre.

5. Sélectionnez le fichier KwrBearingDesignTable.xls et cliquez sur **Ouvrir**. Cliquez sur **Oui** lorsque vous êtes invité à préciser si vous voulez associer les colonnes des tables aux paramètres.

6. Cliquez sur **OK** pour appliquer la configuration par défaut.

7. Enregistrez le fichier, puis fermez-le.

8. Accédez au répertoire contenant le fichier KwrBearingDesignTable.xls et supprimez ce dernier.

9. Revenez dans Catia. Ouvrez le fichier KwrBallBearing1.CATPart : une boîte de dialogue demandant si vous souhaitez sélectionner un nouveau fichier apparaît. Cliquez sur le bouton **Sélectionner** et sélectionnez un nouveau fichier Excel.

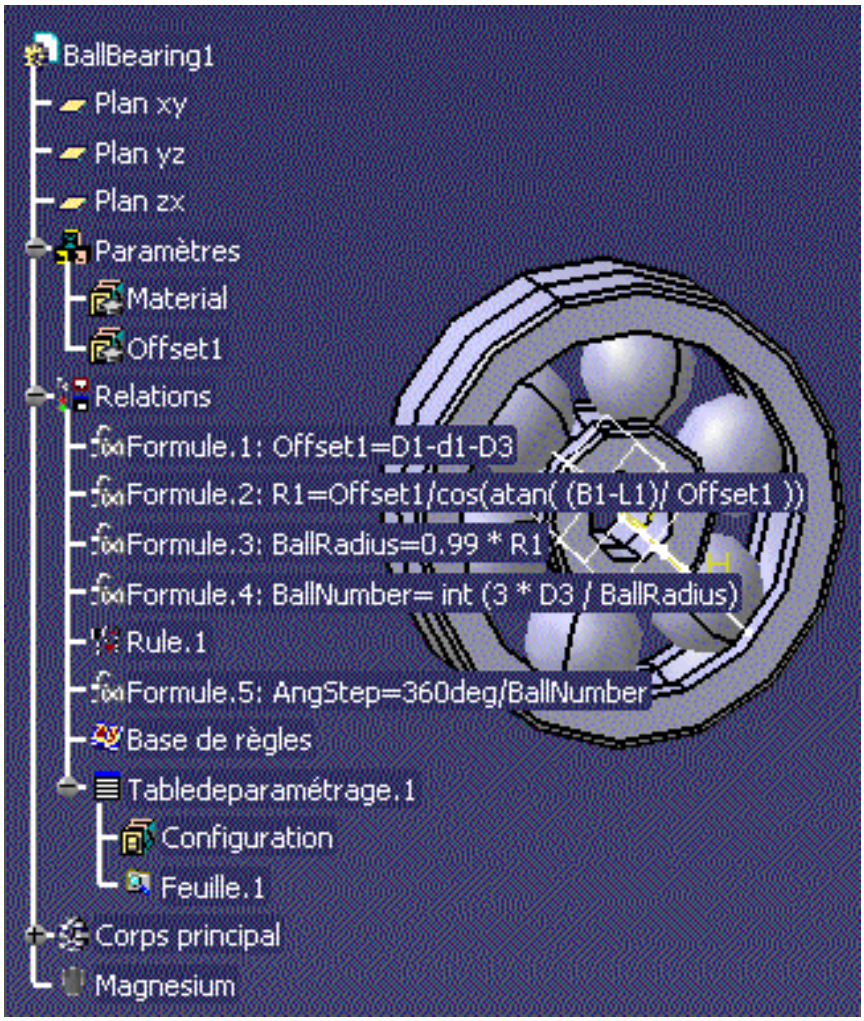
Synchronisation manuelle



Lors du chargement d'un modèle contenant des tables de paramétrage définies par l'utilisateur, si les fichiers correspondants ont été modifiés et que le modèle contient les données des fichiers externes, la table de paramétrage est synchronisée lorsque cette option est sélectionnée. Pour synchroniser les deux fichiers, cliquez à l'aide du bouton droit de la souris sur la table de paramétrage dans l'arbre de spécifications et sélectionnez la commande **DesignTable objet -> Synchroniser** ou **Edition -> Liens**.

1. Dans le menu **Outils -> Options...**, sélectionnez **Général -> Paramètres et mesure** et activez l'option **Synchronisation manuelle** dans l'onglet Knowledge.

2. Ouvrez le fichier KwrBallBearing2.CATPart. Ce fichier contient déjà une table de paramétrage dont les valeurs sont identiques à celles du fichier KwrBearingDesignTable.xls (les fichiers KwrBearingDesignTable.xls et KwrBallBearing2.CATPart doivent se trouver dans le même répertoire).



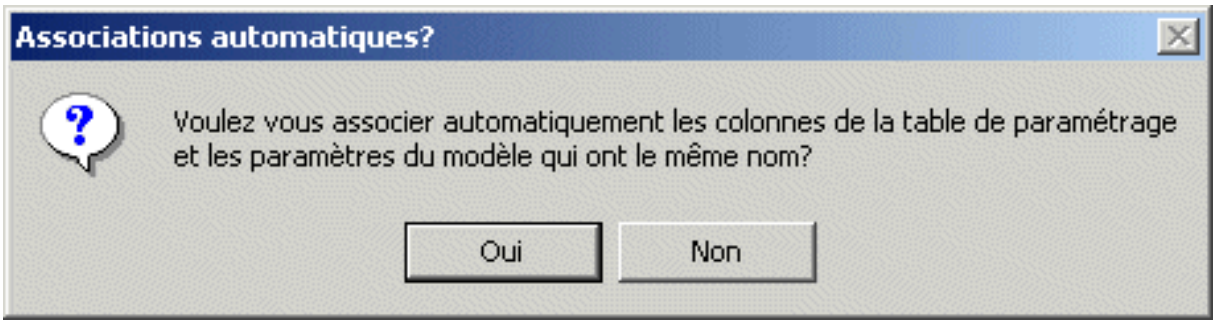
3. Choisissez la commande **Edition -> Liens** pour modifier le chemin du fichier Excel et sélectionner le fichier KwrBearingDesignTable.xls. Enregistrez le fichier, puis fermez-le.

4. Ouvrez le fichier KwrBearingDesignTable.xls et modifiez les valeurs de matériau, par exemple. Fermez le fichier.

5. Revenez dans *CATIA*. Ouvrez le fichier KwrBallBearing2.CATPart.

6. Sélectionnez la commande **Edition -> Liaisons** et cliquez sur le bouton Synchroniser pour synchroniser les deux fichiers.

Si l'option **Dupliquer les données dans le modèle *CATIA*** est active et si vous choisissez un autre fichier de table de paramétrage sans passer par la commande **Modifier la table** dans la session, le message suivant apparaît, quels que soient les paramètres :



Si l'option **Dupliquer les données dans le modèle *CATIA*** est désactivée, la synchronisation se produit de façon automatique.

Stockage d'une table de paramétrage dans une copie optimisée

Dans cette tâche, vous apprendrez à stocker une table de paramétrage dans une copie optimisée en vue de son utilisation ultérieure. Dans ce scénario, l'utilisateur veut instancier dans un contexte différent les cages internes et externes d'un roulement à billes. Pour ce faire, il crée une copie optimisée ne contenant que les cages internes et externes d'un roulement à billes existant.

Le scénario se compose des étapes suivantes :

- Insertion de la table de paramétrage dans le fichier CATPart
- Création de la copie optimisée
- Instanciation de la copie optimisée contenant la table de paramétrage

Pour effectuer ce scénario, l'utilisateur doit posséder la licence Product Knowledge Template.

Pour effectuer ce scénario, l'utilisateur a besoin des fichiers suivants :

- KwrBallBearing1.CATPart
- KwrBearingDesignTable.xls

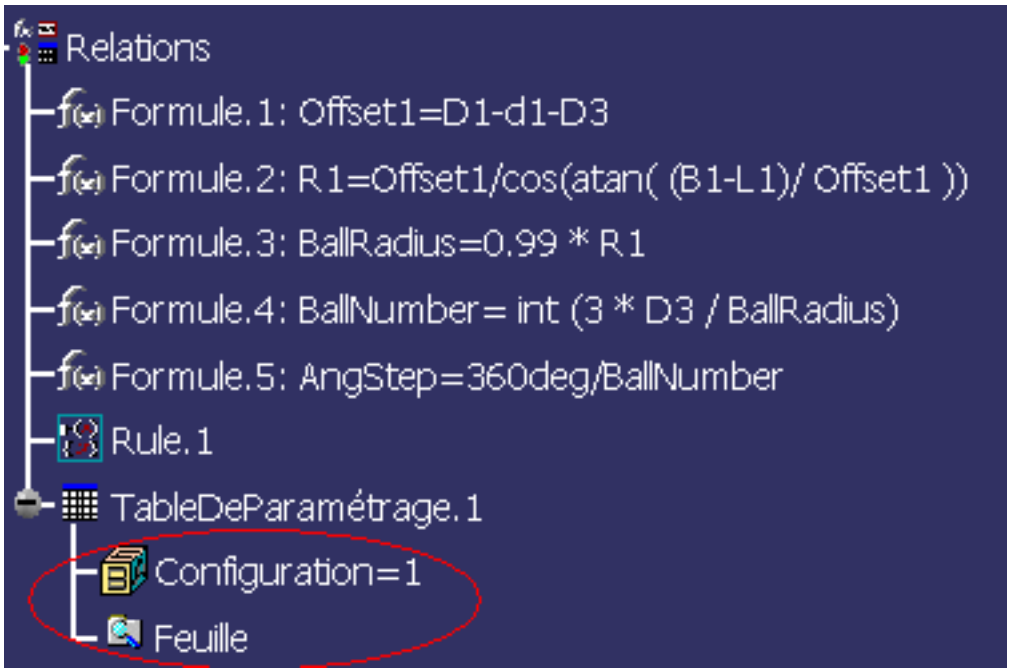
Pour stocker une table de paramétrage dans une copie optimisée, n'oubliez pas de sélectionner les paramètres indiqués par la table de paramétrage.

1. Ouvrez le fichier KwrBallBearing1.CATPart. L'image suivante s'affiche :



Insertion de la table de paramétrage dans le fichier CATPart

2. Cliquez sur l'icône **Table de paramétrage** ( ) dans la barre d'outils standard. La boîte de dialogue **Création d'une table de paramétrage** s'affiche.
3. Activez l'option **Créer une table de paramétrage à partir d'un fichier existant** et cliquez sur **OK**. La boîte de dialogue **Sélection de fichier** s'affiche.
4. Sélectionnez le fichier KwrBearingDesignTable.xls et cliquez sur **Ouvrir**.
5. Cliquez sur **Oui** pour confirmer les associations automatiques, puis cliquez sur **OK**. La table de paramétrage s'affiche désormais sous le noeud Relations.



Création de la copie optimisée

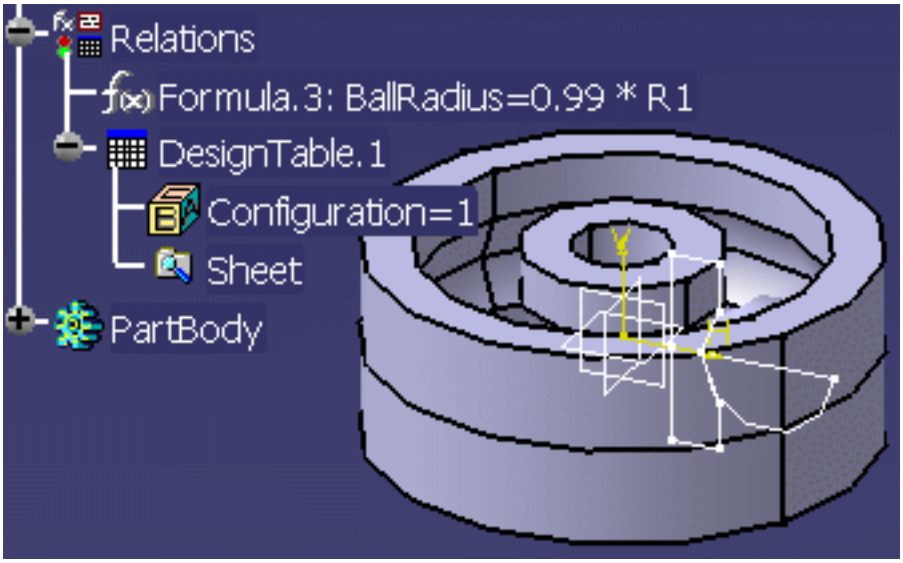
6. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier **Product Knowledge Template** (si besoin est) et cliquez sur l'icône **Création de copie optimisée**. La boîte de dialogue de définition de la copie optimisée s'affiche.
7. Dans l'arbre de spécifications, sélectionnez les composants suivants :
   - Table_paramétrage.1
   - Révolution.1
   - Révolution.2
   - Révolution.3
   - Esquisse.1
   - Esquisse.2
   - Esquisse.3
   - le paramètre Matériau.

- Lorsque vous avez terminé, cliquez sur **OK**. La copie optimisée s'affiche sous le noeud Copies optimisées dans l'arbre de spécifications.

8. Enregistrez le fichier, puis fermez-le.

Instanciation de la copie optimisée

9. Dans le menu **Fichier -> Nouveau**, sélectionnez **Part** dans **Liste des types** et cliquez sur **OK**.
10. Si besoin est, dans le menu **Démarrer -> Knowledgeware**, ouvrez l'atelier **Product Knowledge Template** et cliquez sur l'icône **Instanciation**. La boîte de dialogue **Sélection de fichier** s'affiche.
11. Sélectionnez le fichier KwrBallBearing1.CATPart et cliquez sur **Ouvrir**. La boîte de dialogue **Insertion d'objet** s'affiche.
12. Sélectionnez le plan yz dans l'arbre de spécifications et cliquez sur **OK**. La table de paramétrage est instanciée.



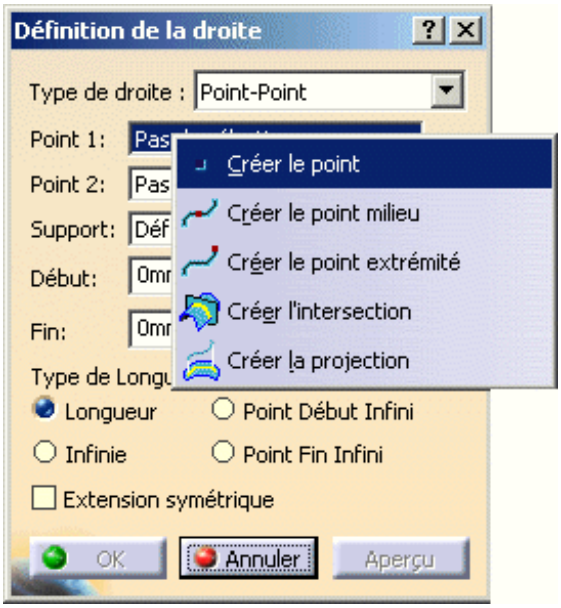
Création et utilisation d'une loi Knowledge Advisor

Le scénario développé ci-après illustre comment créer une loi Knowledge Advisor et l'utiliser avec une loi Generative Shape Design dans la même relation.

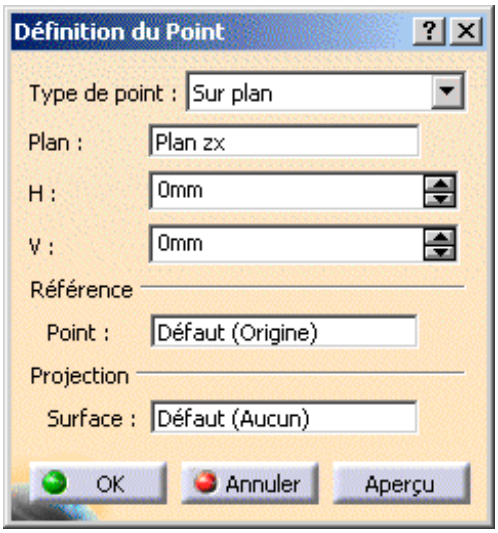
- Lorsqu'un paramètre est défini par une loi Generative Shape Design, vous devez calculer sa valeur par le biais d'une évaluation.
- Notez que le résultat obtenu à la fin de cette tâche dépend des droites de départ. Vous pouvez refaire le scénario avec différentes droites et voir comment cela affecte le résultat.

- Une loi Knowledge Advisor est une relation servant à définir un paramètre par rapport à un autre. Les deux paramètres inclus dans une loi sont appelés les *paramètres formels*. Les paramètres formels et les lois sont spécialement conçus pour être utilisés lors de la création de courbes parallèles Shape Design. Une loi Generative Shape Design peut s'utiliser dans une loi Knowledge Advisor.
- Les lois définissent uniquement une relation entre un paramètre et un autre paramètre unique.

1. Dans le menu **Démarrer->Shape**, ouvrez l'atelier **Generative Shape Design**.
2. Définissez un support de travail à l'aide de l'icône **Travail sur support** ( ).
3. Sélectionnez le plan zx, par exemple, puis cliquez sur **OK** dans la boîte de dialogue Travail sur support mise à jour sans modifier aucun autre paramètre.
4. Cliquez sur l'icône Droite ( ). La boîte de dialogue Droite s'affiche.
5. Cliquez avec le bouton droit dans le champ Point 1 et sélectionnez la commande **Créer un point**.



6. La boîte de dialogue Définition du point s'affiche ; les champs **Type de point** et **Plan** se remplissent automatiquement.
7. Créez un point en affectant les valeurs **H:0mm** et **V:0mm**, puis cliquez sur **OK**.



8. Répétez l'opération en cliquant avec le bouton droit sur le champ Point 2 à partir de la boîte de dialogue Droite pour créer un autre point ayant les valeurs **H:100mm** et **V:0mm**, puis cliquez sur **OK** dans la boîte de dialogue Définition du point.

9. Cliquez sur **OK** dans la boîte de dialogue Droite pour créer la droite.

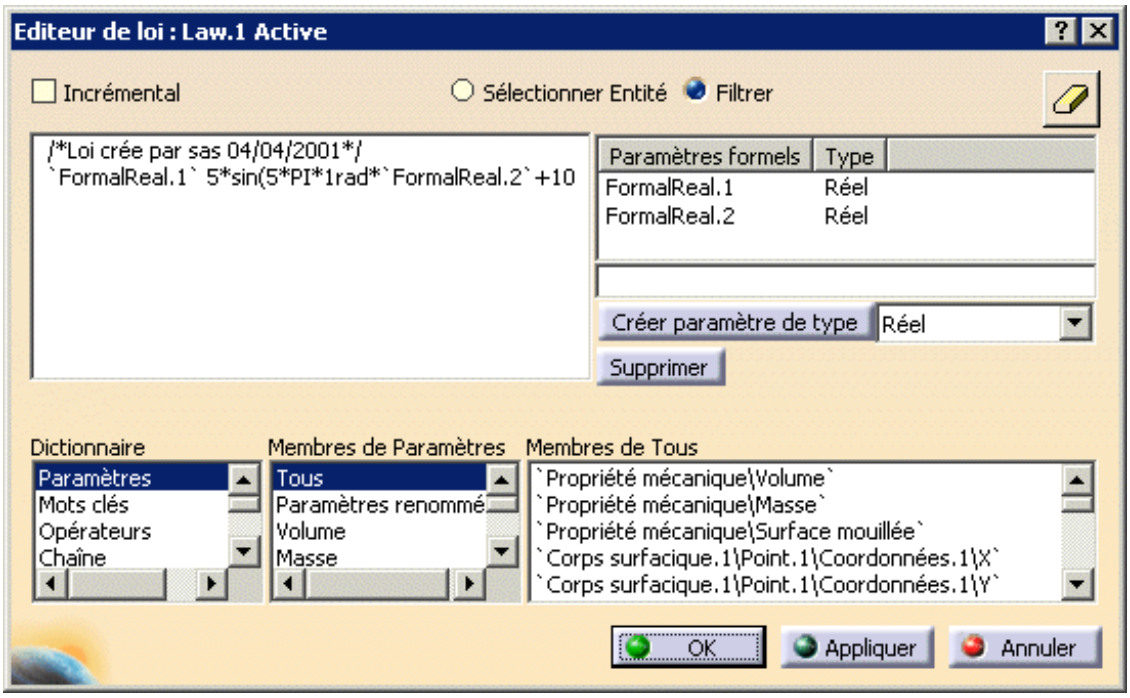
10. Ouvrez l'atelier Knowledge Advisor et cliquez sur l'icône  . Si besoin est, utilisez la commande **Outils->Personnaliser** pour accéder à l'icône. Une boîte de dialogue similaire à celle présentée ci-après s'affiche. Cet éditeur ressemble aux autres éditeurs de relations. Le cas échéant, remplacez les valeurs par défaut dans les champs de la boîte de dialogue.



11. Cliquez sur **OK**. L'éditeur de lois s'affiche. La partie droite permet de créer les paramètres à utiliser dans la loi. La partie gauche est la zone d'édition de la loi.

12. Cliquez sur le bouton **Créer paramètre de type** pour créer deux paramètres de type réel FormalReal.1 et FormalReal.2, puis entrez la loi ci-après dans la fenêtre d'édition :

    FormalReal.1 = 5*sin(5*PI*1rad*FormalReal.2)+ 10

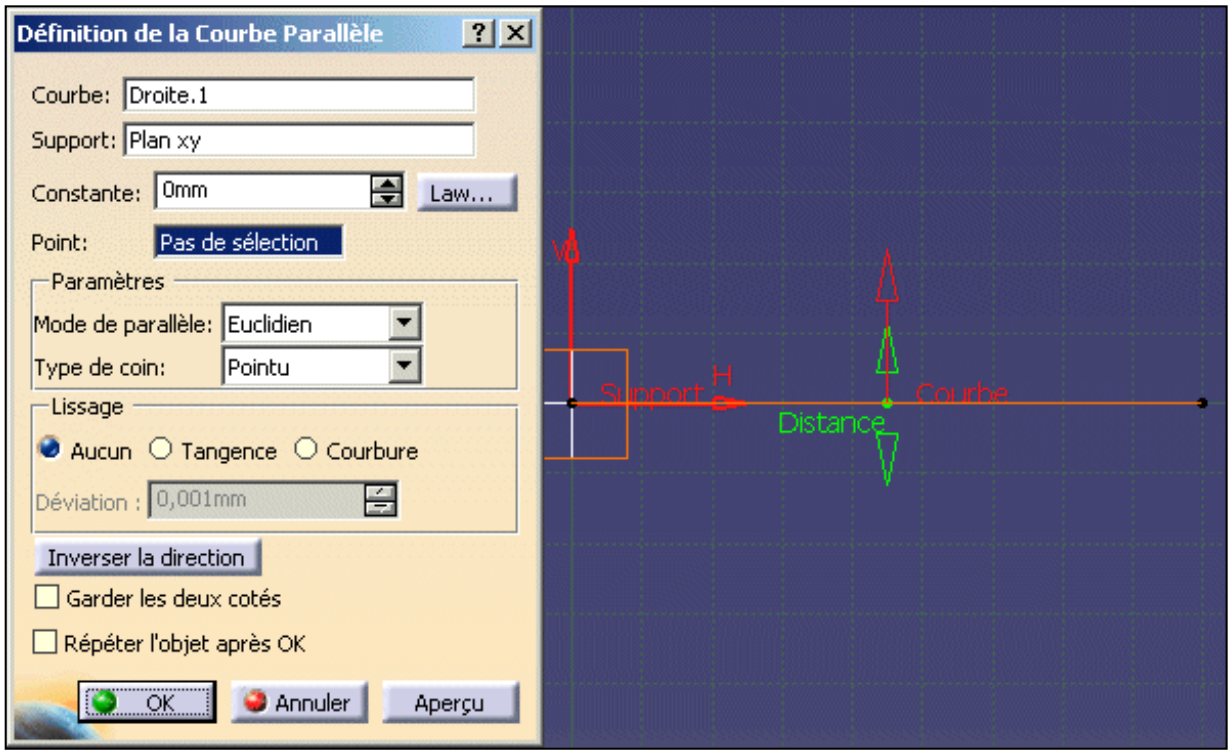


13. Cliquez sur **OK** pour ajouter la loi au document. L'entité Law.1 est ajoutée à l'arbre de spécifications juste en dessous du noeud Relations.

14. Sélectionnez l'entité racine du document et rouvrez l'atelier Generative Shape Design.

15. Cliquez sur l'icône  pour créer une courbe parallèle à la droite créée au tout début du scénario. La boîte de dialogue **Définition de la courbe parallèle** s'affiche.

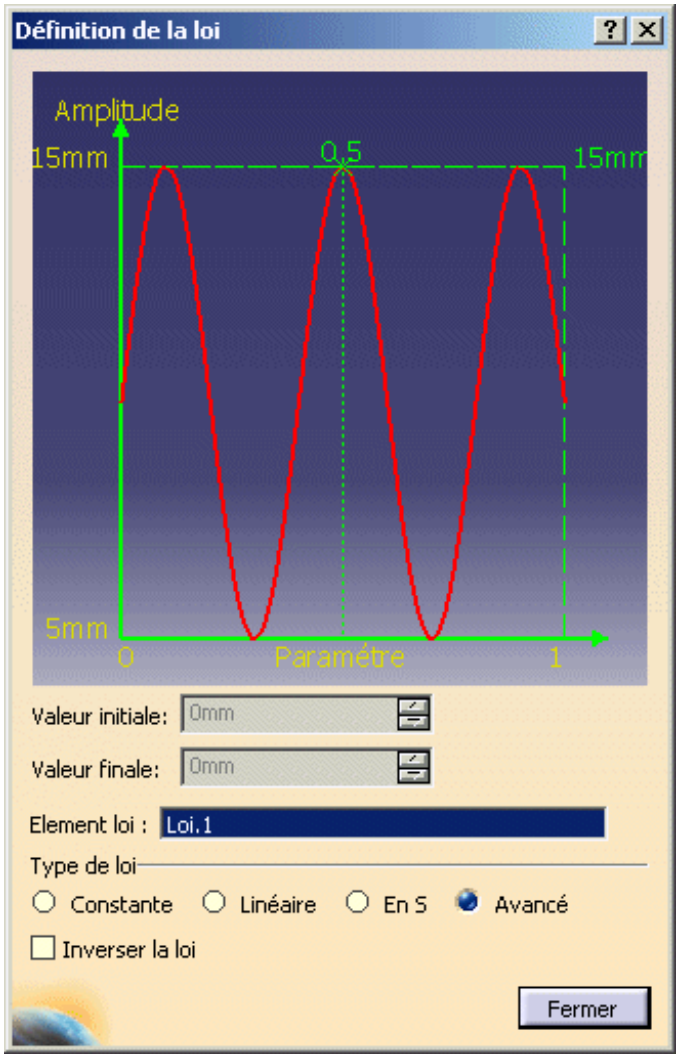
16. Sélectionnez la droite précédemment créée en tant que **courbe** de référence.



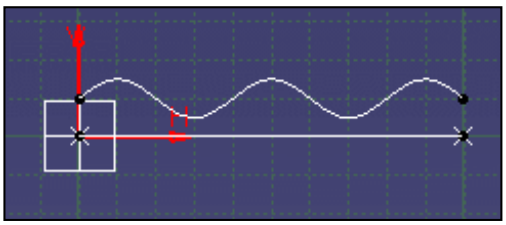
Notez que vous ne pouvez utiliser que des lois positives, à savoir des lois ayant des valeurs positives, pour créer des courbes parallèles (le terme "positives" est à comprendre dans le sens "strictement positives").

18. Cliquez sur le bouton **Loi...**. La boîte de dialogue Définition de la loi s'affiche.

19. Sélectionnez le type de loi **Avancé**, cliquez sur Law.1 dans l'arbre de spécifications, puis sur **Fermer**.



20. Cliquez sur **OK**. Une courbe parallèle à la courbe sélectionnée est créée en tenant compte de la loi.



L'exemple KwrCreatingaLaw.CATPart illustre ce scénario.

Utilisation de l'inspecteur de gestion de connaissance

L'inspecteur de gestion de connaissance permet d'identifier et de prévisualiser les résultats de la modification des paramètres d'un document sans que ces modifications ne soient effectivement mises en oeuvre. Cette "analyse des dépendants" donne un aperçu immédiat des résultats à venir et permet d'affiner la conception des documents créés.

S'il est important de pouvoir déterminer l'effet de la modification d'un ou de plusieurs paramètres, il est également souhaitable de pouvoir déterminer comment modifier son document pour obtenir le résultat désiré. L'inspecteur de gestion de connaissance répond à ce souhait en vous permettant de demander "comment" réaliser une modification donnée.

En somme, l'inspecteur de gestion de connaissance est un outil qui permet d'étudier les effets et les conditions requises des modifications.

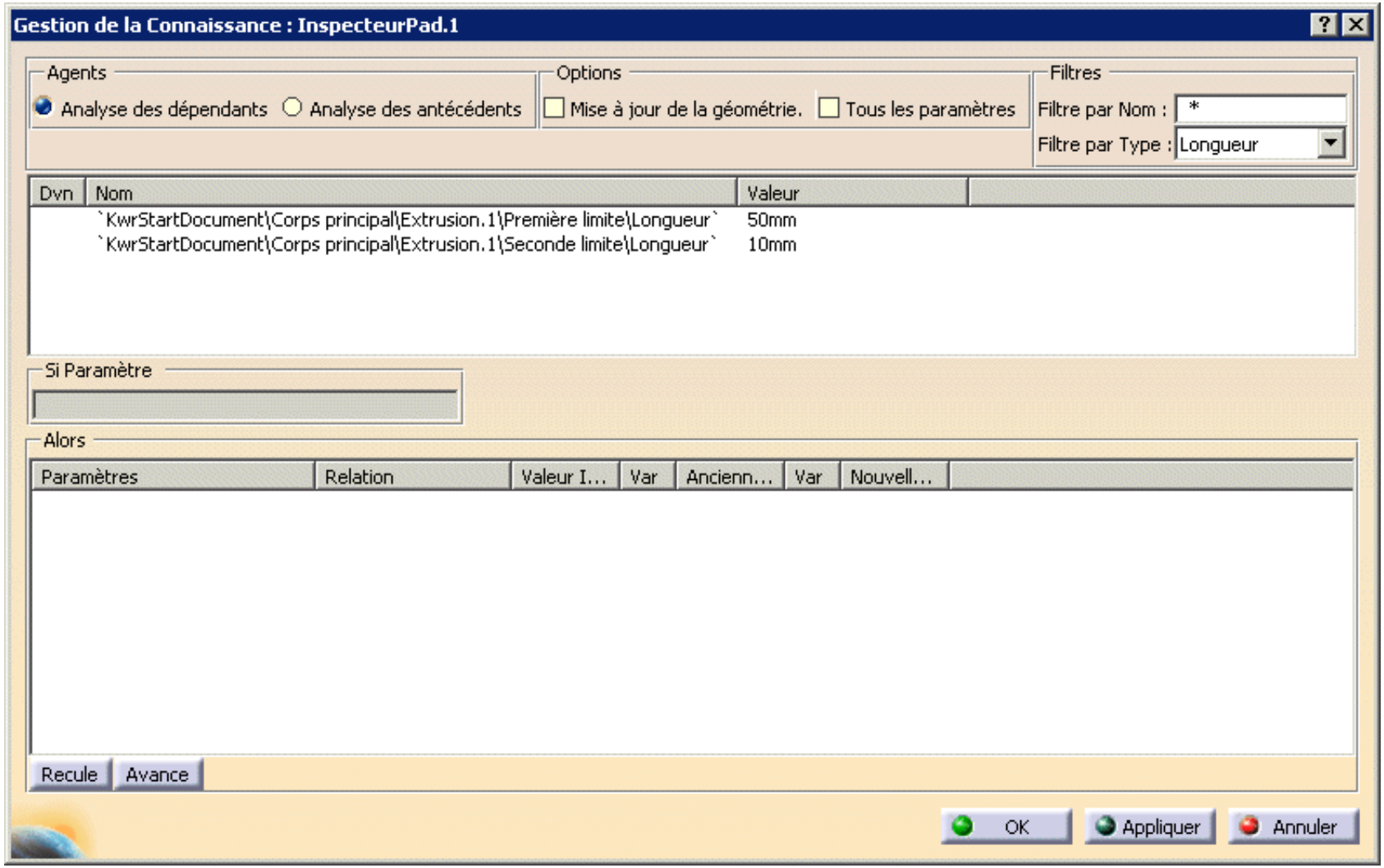
Analyse des dépendants (*effets*)	La fonction Analyse des dépendants permet de comprendre comment la modification d'un paramètre du document (qu'il s'agisse du matériau, de la pression ou d'une cote) modifie le fonctionnement ou la conception du produit sur lequel vous travaillez. Elle permet d'examiner l'interaction existant entre les paramètres et avec les règles qui composent les spécifications du produit. L'option Mise à jour de la géométrie permet de visualiser le résultat des modifications dans la zone géométrique.
Analyse des antécédents *(dépendances)*	L'option Analyse des antécédents permet de déterminer comment modifier le produit pour obtenir un résultat donné.

N'utilisez pas les fonctions avec l'inspecteur de gestion de connaissance.

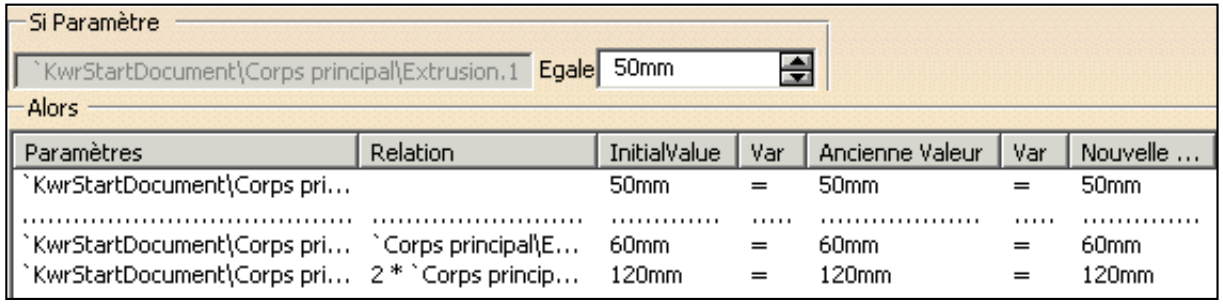
Mode Analyse des dépendants

Dans cette tâche, vous apprendrez à utiliser le mode Analyse des dépendants.

1. Ouvrez le document KwrFormula1.CATPart ainsi que l'atelier Knowledge Advisor.
2. Cliquez sur l'icône de l'inspecteur de gestion de connaissance ou sélectionnez-le dans la barre d'outils standard. La boîte de dialogue de l'inspecteur de gestion de connaissance s'affiche. Sélectionnez l'option Analyse des dépendants.
3. Sélectionnez le paramètre KwrStartDocument\ CorpsPrincipal\Extrusion.1 \PremièreLimite \Longueur (à ce stade, ne modifiez pas sa valeur dans le champ Est égal à).

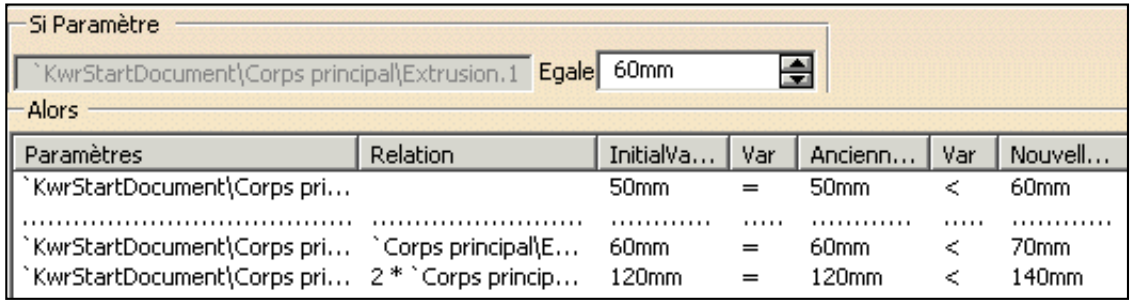


4. Cliquez sur **Appliquer**. La liste de paramètres et de valeurs suivante s'affiche dans la zone Alors.



La première ligne décrit le dernier paramètre sélectionné. Les autres lignes décrivent les paramètres dépendants.

5. Utilisez le champ Est égal à pour remplacer la valeur du paramètre KwrStartDocument\ CorpsPrincipal\Extrusion.1 \PremièreLimite \Longueur par 60 mm. Cliquez sur **Appliquer**. Dans la zone Alors, les valeurs des paramètres sont mises à jour de la manière suivante :



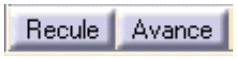
Paramètres	Relation	InitialVa...	Var	Ancienn...	Var	Nouvell...
`KwrStartDocument\Corps pri...		50mm	=	50mm	<	60mm
.....	.....	.....	.....	.....	.....	.....
`KwrStartDocument\Corps pri...	`Corps principal\E...	60mm	=	60mm	<	70mm
`KwrStartDocument\Corps pri...	2 * `Corps princip...	120mm	=	120mm	<	140mm

La colonne Valeur initiale indique les valeurs initiales des paramètres (à l'ouverture de l'inspecteur de gestion de connaissance). La colonne Ancienne valeur contient les valeurs des paramètres issues de l'opération Analyse des dépendants précédente. Les colonnes Var (variations) contiennent les opérateurs de comparaison utilisés entre les valeurs des colonnes adjacentes.

6. Sélectionnez l'option **Mise à jour géométrique** pour afficher les modifications issues de l'opération Analyse des dépendants dans la zone géométrique. Cliquez sur **Appliquer** pour mettre à jour le document dans la zone géométrique.

7. Cliquez sur **OK** pour appliquer au document les valeurs produites par l'opération Analyse des dépendants courante. Si vous ne voulez pas appliquer ces valeurs, cliquez sur Annuler>.

Remarques :

- Utilisez les boutons 

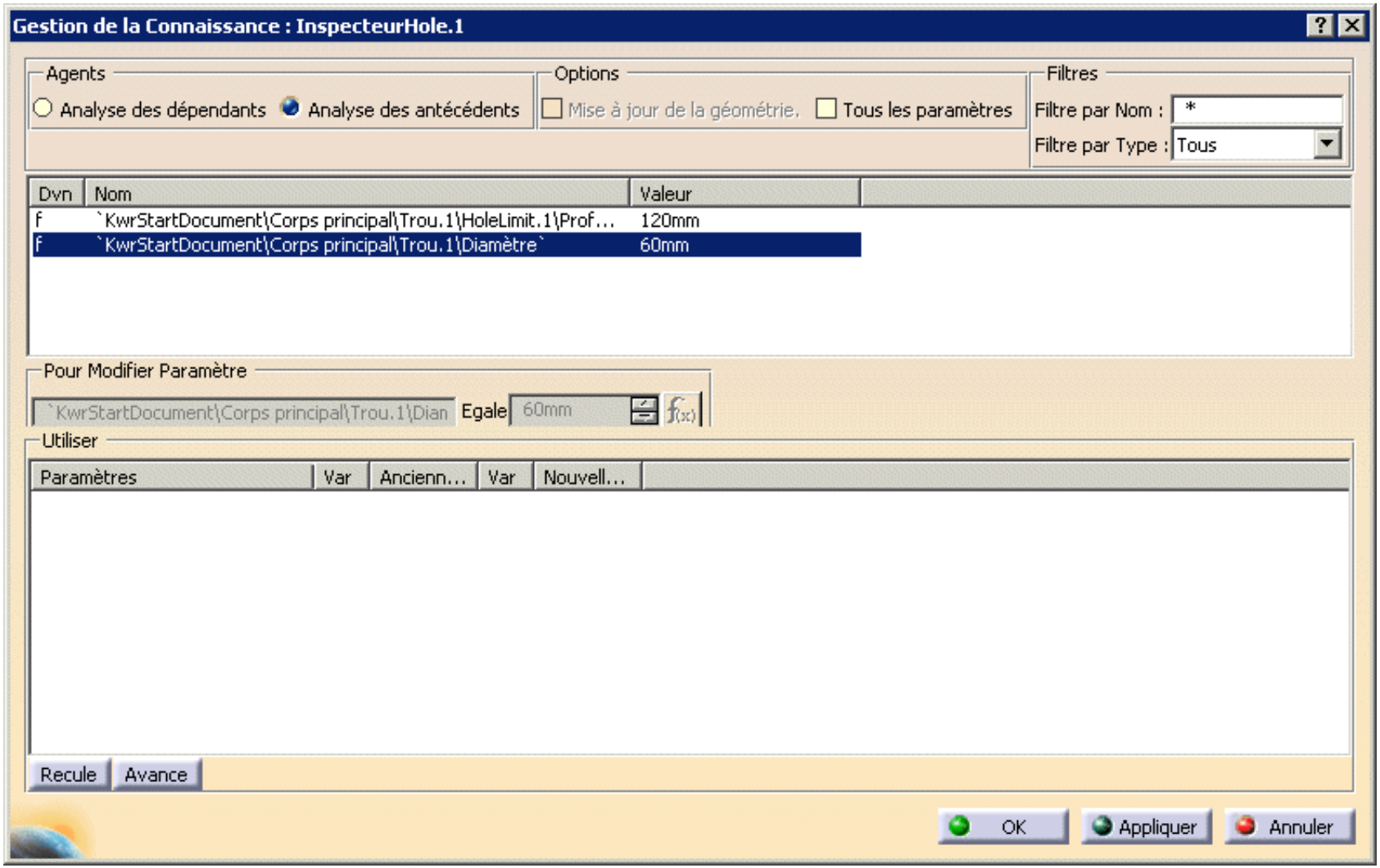
pour insérer dans la zone 'Alors' les valeurs précédentes ou suivantes dans l'historique des opérations 'Analyse des dépendants'.
- Pour afficher tous les paramètres du document, sélectionnez l'option **Tous les paramètres**. La lettre `f`, dans la colonne Dvn, indique que le paramètre est assorti d'une formule de contrainte.
- Si l'option **Tous les paramètres** a été cochée et que vous sélectionnez un paramètre dans la zone **Alors**, alors le paramètre choisi est mis en évidence dans la liste placée au dessus.

La modification d'une valeur de paramètre n'entraîne pas la mise à jour automatique des paramètres dépendants en cas d'opération Analyse des dépendants. Par exemple, si un paramètre est assorti d'une formule de contrainte, telle que :
if Paramètre1 > A then Paramètre2 = B
remplacer la valeur *Paramètre1* par une valeur supérieure à A ne modifiera pas la valeur de Paramètre2 si Paramètre2 avait initialement B pour valeur.

Mode Analyse des antécédents

Dans cette tâche, vous apprendrez à utiliser le mode Analyse des antécédents.

1. Ouvrez le document KwrFormula1.CATPart.
2. Cliquez sur l'icône de l'inspecteur de gestion de connaissance  dans la barre d'outils standard. La boîte de dialogue Inspecteur de gestion de connaissance s'affiche. Cochez l'option **Analyse des antécédents**. Par défaut, seuls les paramètres assortis d'une formule de contrainte sont affichés.
3. Le cas échéant, cochez l'option **Tous les paramètres** pour afficher tous les paramètres du document.
4. Sélectionnez le paramètre KwrStartDocument\ CorpsPrincipal\ Trou.1\Diamètre (imaginons que vous souhaitez modifier ce paramètre).
5. Cliquez sur **Appliquer** ou sur **Entrée**. La liste des paramètres à modifier pour changer la valeur du paramètre Trou.1\Diamètre s'affiche dans la zone Utiliser.
6. Sélectionnez le paramètre Extrusion.1\PremièreLimite\Longueur.



7. Sélectionnez l'option **Analyse des dépendants**.

8. Modifiez le paramètre PremièreLimite\Longueur en mode Analyse des dépendants.

9. Cliquez sur OK pour appliquer au document la modification du paramètre.

Remarques :

- Pour afficher tous les paramètres du document, sélectionnez l'option Tous les paramètres. La lettre `f`, dans la colonne Dvn, indique que le paramètre est assorti d'une formule de contrainte.
- Si l'option **Tous les paramètres** a été cochée et que vous sélectionnez un paramètre dans la zone Utiliser, alors le paramètre choisi est mis en évidence dans la liste placée au dessus.



Utilisation de la fonction Règle



Cliquez sur l'icône Règle pour créer une règle, écrire son code, valider sa syntaxe et l'appliquer à un document.

Une règle est une série d'instructions, généralement conditionnelles, qui permet de contrôler les relations entre les paramètres. De plus, selon le contexte défini par les instructions de la règle, des actions peuvent être exécutées :

- pour définir une valeur ou une formule selon des paramètres, relatifs à l'activité des entités par exemple ;
- pour afficher des fenêtres d'information ;
- pour lancer les macros Visual Basic stockées dans des fichiers externes ou dans le document V5 ;
- pour affecter des points, des courbes et des surfaces et par conséquent permettre des modifications topologiques contextuelles et automatiques.

Dans l'arbre de spécifications, la règle apparaît sous la forme d'une relation qui peut être activée ou désactivée. Comme n'importe quelle entité, une règle peut être manipulée à partir de son menu contextuel.

Une règle est exécutée lorsque l'un de ses paramètres d'entrée ou l'une de ses entités d'entrée a changé, et si vous demandez la mise à jour de la règle.

Il en découle que vous ne pouvez pas décider entièrement du moment où la règle doit être déclenchée. C'est pourquoi les règles ne doivent manipuler que des paramètres et des entités, et doivent être utilisées pour contrôler le statut d'une conception (modification de paramètres et de la géométrie).



Si vous souhaitez décider du moment où l'action sera exécutée, vous devez utiliser la fonction Réaction.

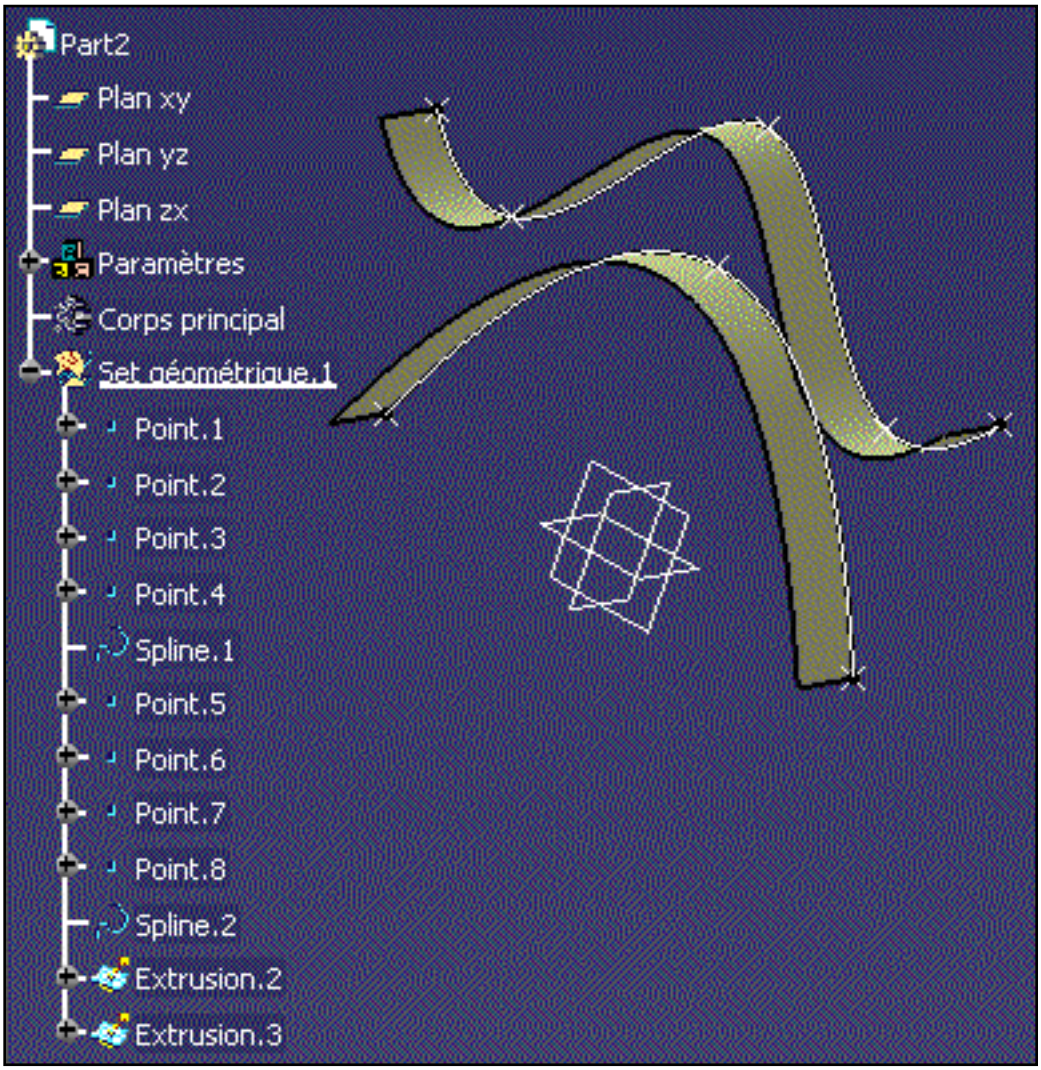
Création d'une règle
Utilisation de l'éditeur de règles
Utilisation de règles et de vérifications dans une copie optimisée
Utilisation du dictionnaire

Création d'une règle

Dans cette tâche, vous apprendrez à créer une règle qui récupère l'abscisse d'un point et, selon la valeur des coordonnées, affiche un message donné.

Cet exemple utilise deux fonctions spécifiques permettant d'extraire les coordonnées d'un point. Vous pouvez accéder à ces fonctions à partir de l'élément Mesures du dictionnaire.
Pour plus d'informations sur le dictionnaire, voir Utilisation du dictionnaire.

1. Ouvrez le document KwrMeasure.CATPart. Le document a été entièrement créé avec Generative Shape Design. Les surfaces extrudées sont extrudées à partir des courbes Spline.1 et Spline.2. Le point dont les coordonnées sont à extraire et à tester est Point.5.



2. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier **Knowledge Advisor**.

3. Utilisez l'éditeur  pour créer trois paramètres de type Longueur : Point5X, Point5Y et Point5Z.

4. Cliquez sur l'icône  . Dans la première boîte de dialogue qui s'affiche, tapez le nom d'une règle (RègleMesure, par exemple). Le cas échéant, remplacez les commentaires par défaut. Si vous ajoutez la règle à créer à un ensemble de relations spécifique, indiquez une cible. Pour ce faire, reportez-vous à la section Création d'ensembles de relations.

5. Cliquez sur OK. L'éditeur de règles s'affiche.

6. Entrez la règle ci-dessous dans la fenêtre d'édition.

```
if Geometrical Set.1\Point.5.coord(1) > 0mm
Message("L'abscisse de Point.5 est positive")
else
{
 Geometrical Set.1\Point.5.coord(Point5X, Point5Y, Point5Z)
 Message("L'abscisse de Point.5 est # ", Point5X)
}
```

7. Dans la règle ci-dessus, vous pouvez extraire la définition du Point.5 (Geometrical Set.1\Point.5). Pour cela, cliquez deux fois sur l'entité dans l'arbre de spécifications.

8. Cliquez sur **OK**. Le message "L'abscisse de Point.5 est : 0 mm" s'affiche.

9. Modifiez l'entité Point.5 (cliquez deux fois sur l'objet dans l'arbre de spécifications par exemple) et remplacez la valeur X du Point.5 par 10 mm. La règle est à mettre à jour. Pour plus d'informations sur les relations à mettre à jour, reportez-vous à la section Mise à jour des mesures.

10. Ouvrez à nouveau l'atelier Knowledge Advisor, puis cliquez sur l'icône  . Un message vous informe que "l'abscisse du Point.5 est positive".

Pour plus d'informations sur l'éditeur de règles, voir Utilisation de l'éditeur de règles.



Utilisation de règles et de vérifications dans une copie optimisée

Dans cette tâche, vous apprendrez à utiliser des règles et des vérifications dans une copie optimisée.

Les règles et les vérifications ainsi que d'autres relations peuvent être appliquées à un document en les récupérant dans un autre document, à condition qu'elles aient été stockées dans une *copie optimisée*. Pour plus d'informations sur les mécanismes de copie optimisée, reportez-vous au document *Generative Shape Design - Guide de l'utilisateur*.

1. Ouvrez le document KwrMeasurePCopy.CATPart. Le cas échéant, ouvrez l'atelier Generative Shape Design.
2. Dans la barre de menus standard, sélectionnez la commande **Insertion->Knowledge Templates->Copie optimisée...**. La fenêtre Définition de copie optimisée s'affiche.
3. Dans l'arbre de spécifications, sélectionnez les relations Règle.1 et Vérification.1. Ces deux relations sont reportées sur la fenêtre Définition de copie optimisée. Cliquez sur **OK** dans la fenêtre Création de copie optimisée. Enregistrez et quittez le document.
4. Ouvrez le document KwrSplineInPcopy1.CATPart et l'atelier Generative Shape Design.
5. Sélectionnez la commande **Insertion->Instanciation...** dans la barre de menus standard. La boîte de dialogue Sélection de copie optimisée s'affiche. Sélectionnez le document contenant la copie optimisée où sont stockées les relations Règle.1 et Vérification.1, puis cliquez sur **Ouvrir**. La boîte de dialogue **Insertion d'objet** s'affiche.
6. Sélectionnez l'entité Spline.1 dans l'arbre de spécifications ou dans la zone géométrique. Cliquez sur **OK**. Un message généré par la vérification vous informe que la longueur de la spline est <100 mm. Les deux relations sont reportées dans l'arbre de spécifications et l'icône de vérification est rouge. La règle n'a pas été déclenchée.

7. Ouvrez le document KwrSplineInPcopy2.CATPart et recommencez la même opération (à partir de l'étape 5). La longueur de la spline apparaît dans une boîte d'information indiquant que la règle Règle.1 est déclenchée. Cette fois, l'icône de vérification devient verte dans l'arbre de spécifications.

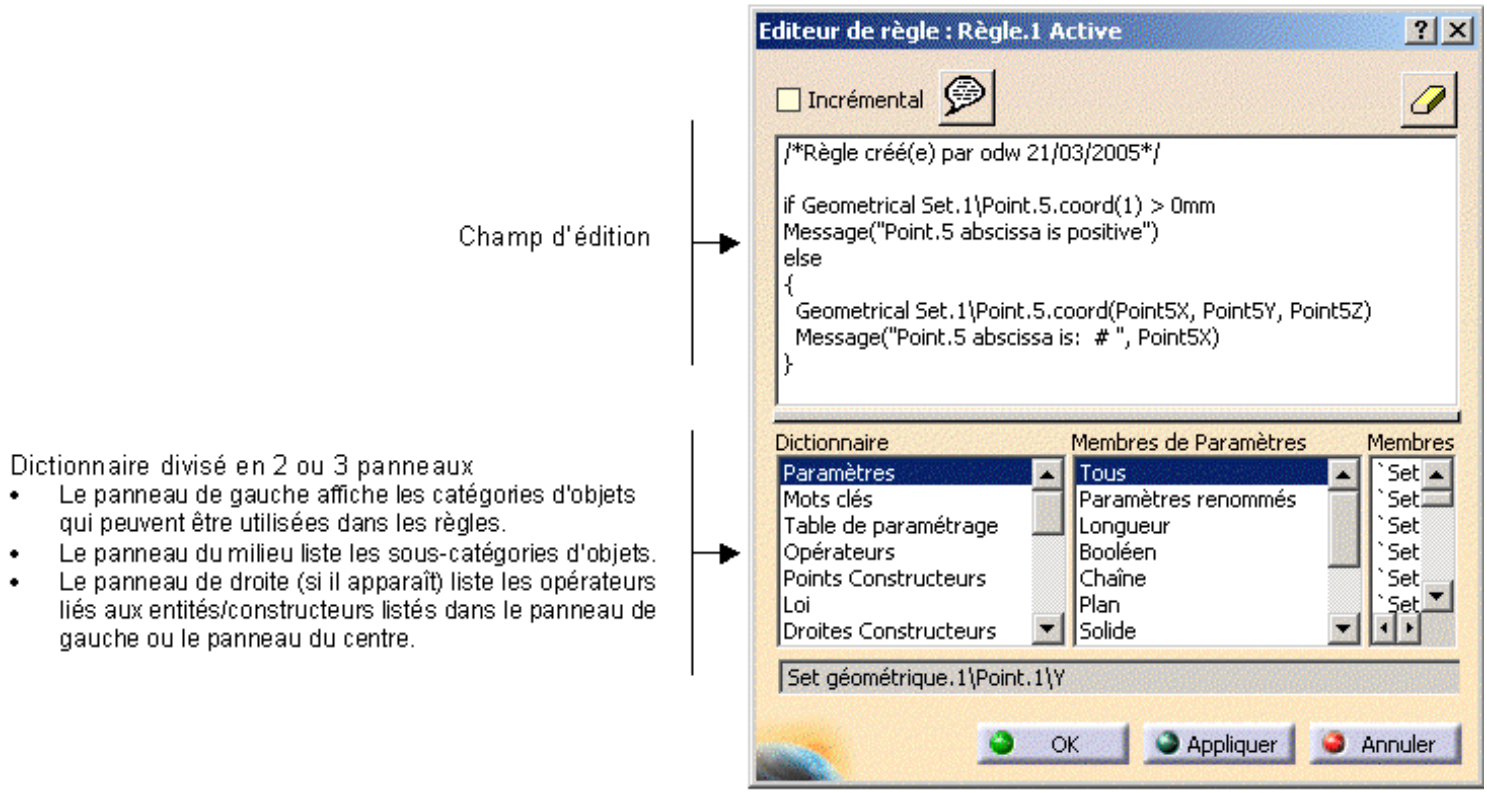


Les règles et les vérifications peuvent être stockées dans des catalogues et instanciées ultérieurement dans un document. Reportez-vous à la section Instanciation des relations de règles de connaissance issues d'un catalogue.



Utilisation de l'éditeur de règles

L'éditeur de règles est destiné à vous aider à entrer le corps de la vérification à l'aide du dictionnaire. Il se compose des éléments suivants :



Dans l'éditeur de règles, vous pouvez :

- Limiter la liste des paramètres affichés dans le dictionnaire. Il suffit de cliquer dans l'arbre de spécifications sur l'entité dont vous souhaitez afficher les paramètres. Si la case Incrémental est cochée, seuls les paramètres du premier niveau sont affichés juste en dessous de l'entité sélectionnée, sinon tous les paramètres de tous les niveaux apparaissent. Supposez que votre document contienne une entité Geometrical Set elle-même composée de plusieurs points Shape Design. Si la case Incrémental est décochée et que l'entité Geometrical Set est sélectionnée dans l'arbre de spécifications, tous les paramètres associés aux points (les paramètres qui définissent les coordonnées sont répertoriés dans la liste) seront affichés. Si la case Incrémental est cochée et que l'entité Geometrical Set est sélectionnée, seul le premier niveau des paramètres apparaîtra sous l'entité Geometrical Set (les coordonnées des points ne sont pas affichés).
- Insérer la définition de l'entité dans une règle. Pour cela, cliquez deux fois dans l'arbre de spécifications sur l'entité dont vous souhaitez insérer la définition.
- Vérifier si la syntaxe de la règle est correcte. Pour cela, cliquez sur **Appliquer**.
- Effacer le contenu de la fenêtre d'édition. Pour cela, cliquez sur l'icône  .
- Ajouter la règle au document. Pour cela, cliquez sur **OK**.

Pour plus d'informations sur les options du dictionnaire, reportez-vous à la section Utilisation du dictionnaire ou sélectionnez l'une des options et appuyez sur la touche F1 dans Catia.
Pour plus d'informations sur les règles, voir Création d'une règle.

Traitement des erreurs

Dans cette tâche, vous apprendrez à traiter les composants défaillants. Dans le scénario présenté ci-dessous, vous allez ouvrir un fichier .CATPart composé de deux droites et d'un point de référence. Ce point formera l'intersection des droites si elles se coupent ou bien l'origine, dans le cas contraire. Pour ce faire, vous allez créer une règle :

L'utilisation d'opérateurs géométriques pour définir les valeurs de la géométrie dans des relations peut conduire à des erreurs de mise à jour dans les composants créés. Si vous définissez un point de référence avec le résultat de l'intersection de 2 droites, ces droites risquent de ne pas se couper et le point sera alors incorrect.

L'option **Ne pas capter les erreurs d'évaluation**, disponible lorsque vous sélectionnez la relation dans l'arbre de spécifications et la commande Propriétés, permet d'effectuer les opérations suivantes :

- créer des composants en erreur ;
- déterminer (par le biais d'un message d'erreur) qu'un composant est erroné et ;
- effectuer les modifications nécessaires en cas d'erreurs.

Notez que cette fonction de traitement des erreurs est disponible uniquement pour les composants GSD.

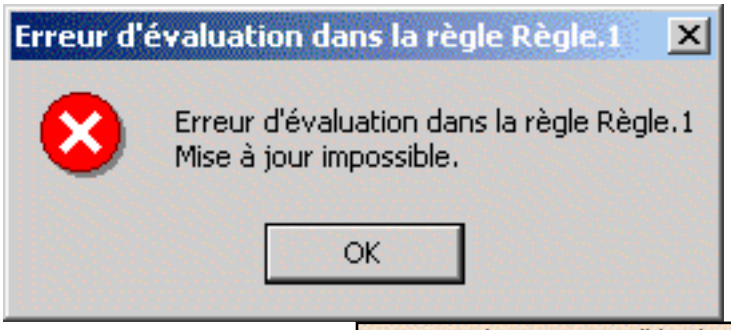
1. Ouvrez le fichier KwrErrorHandling.CATPart.
2. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier Knowledge Advisor.
3. Cliquez sur l'icône Règle () pour créer une règle qui s'appliquera dans la situation suivante : le point de référence constitue le point d'intersection des deux droites si elles se coupent, ou l'origine dans le cas contraire.
4. Dans l'éditeur de règles, entrez le corps suivant :

```
let x(Point)
x =intersect(`Geometrical Set.1\Line.1` ,`Geometrical
Set.1\Line.2` )
if (x.Error==true)
`Geometrical Set.1\Point.5` = point(0mm,0mm,0mm)
else
`Geometrical Set.1\Point.5` = x
```

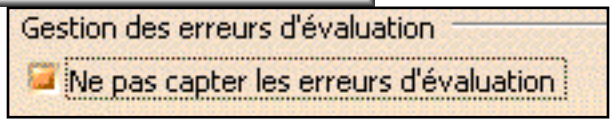
Les utilisateurs japonais doivent entrer le corps de la règle suivant :

```
let x(Point)
x = intersect(`形状セット.1¥直線.1` ,`形状セット.1¥直線.2` )
if (x.Error==true)
`形状セット.1¥点.5` = point(0mm,0mm,0mm)
else
`形状セット.1¥点.5` =x
```

5. Lorsque vous avez terminé, cliquez sur **OK**. L'opérateur d'intersection émet un message d'erreur si l'intersection est vide.



6. Cliquez avec le bouton droit de la souris dans l'arbre de spécifications et sélectionnez la commande **Propriétés**.



7. Cochez la case **Ne pas capter les erreurs d'évaluation** et cliquez sur **OK** lorsque vous avez terminé. De cette manière, l'évaluation de la règle sera forcée, même si une erreur de mise à jour se produit.

Utilisation de la fonction Vérification



Cliquez sur cette icône pour créer une vérification, écrire son code, valider sa syntaxe et l'appliquer à un document.

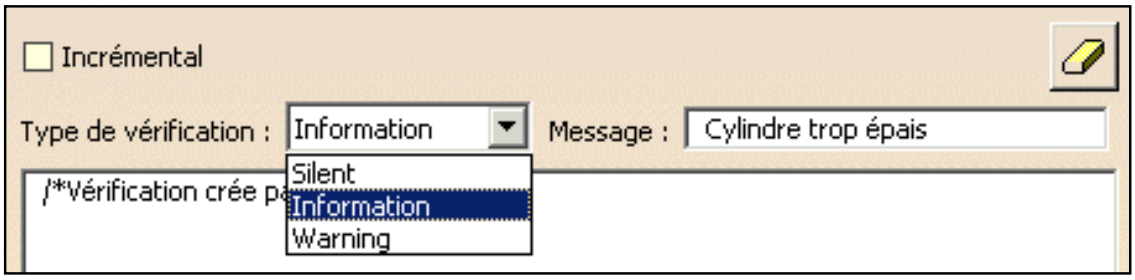
Une vérification est une série d'instructions permettant de savoir si certaines conditions sont remplies ou non. Une vérification ne modifie pas le document auquel elle s'applique. Une vérification est une entité. Dans l'arbre de spécifications du document, elle apparaît sous la forme d'une relation qui peut être activée et désactivée. Comme n'importe quelle entité, une vérification peut être manipulée à partir de son menu contextuel.

Création d'une vérification
Utilisation de l'éditeur de vérifications
Exécution d'une analyse globale des vérifications
Utilisation de l'outil d'analyse des vérifications
Utilisation de règles et de vérifications dans une copie optimisée
Personnalisation des rapports de vérification
Utilisation du dictionnaire

Création d'une vérification

Dans cette tâche, vous apprendrez à créer une vérification.

1. Ouvrez le document KwrFormula0.CATPart, sélectionnez l'élément racine dans l'arbre de spécifications et ouvrez l'atelier Knowledge Advisor.
2. Cliquez sur l'icône  . La première boîte de dialogue de l'éditeur de vérifications s'affiche.
3. Remplacez le nom par défaut par VérifCylindre. Le cas échéant, ajoutez un commentaire dans la zone Description.
   Si vous voulez ajouter la nouvelle vérification à un ensemble de relations spécifique, indiquez une cible. Pour ce faire, reportez-vous à la section Création d'ensembles de relations. Par défaut, la vérification est créée juste en dessous du noeud Relations.
4. Cliquez sur OK. L'éditeur de vérifications s'affiche.



5. Sélectionnez un type de vérification. Entrez le message que vous voulez voir s'afficher dans la fenêtre d'information ou d'avertissement, si la vérification échoue.
6. Entrez les instructions de la vérification dans la fenêtre d'édition. Vous pouvez y insérer les instructions suivantes en faisant un copier/coller.

   ```
   Relations\Formula.1\Activity == false
   ```

7. Cliquez sur **Appliquer** pour tester la validité de la syntaxe. Si le message d'information s'affiche, cela signifie que la syntaxe de vérification est correcte.
8. Cliquez sur **OK** pour ajouter VérifCylindre au noeud Relations dans l'arbre de spécifications. Une icône rouge s'affiche dans l'arbre de spécifications pour indiquer que la vérification a échoué.

9. Désactivez Formula.1, l'icône de vérification devient verte dans l'arbre de spécifications.

Trois paramètres associés à la vérification s'affichent dans la boîte de dialogue "Formules" :

- l'activité ;
- la gravité ;
- le résultat.

Lorsque vous sélectionnez le paramètre de résultat, l'icône indiquant que la vérification a réussi ou échoué s'affiche en face du champ de valeur. Pour ouvrir l'éditeur de vérifications, cliquez deux fois sur cette icône.

Pour plus d'informations sur l'éditeur de vérifications, voir Utilisation de l'éditeur de vérifications.

Pour plus d'informations sur l'éditeur de vérifications, voir Utilisation de l'éditeur de vérifications.



Exécution d'une analyse globale des vérifications

Dans cette tâche, vous apprendrez à exécuter l'analyse des vérifications Knowledge Expert et Knowledge Advisor. L'exemple se divise en deux étapes principales :

- Les paramètres, formules et vérifications sont crées.
- L'analyse des vérifications est exécutée et les vérifications ayant échoué sont corrigées.

Pour plus d'informations sur l'outil d'analyse globale et sur le rapport de vérification, voir Utilisation de l'outil Analyse des vérifications et Personnalisation des rapports de vérification.

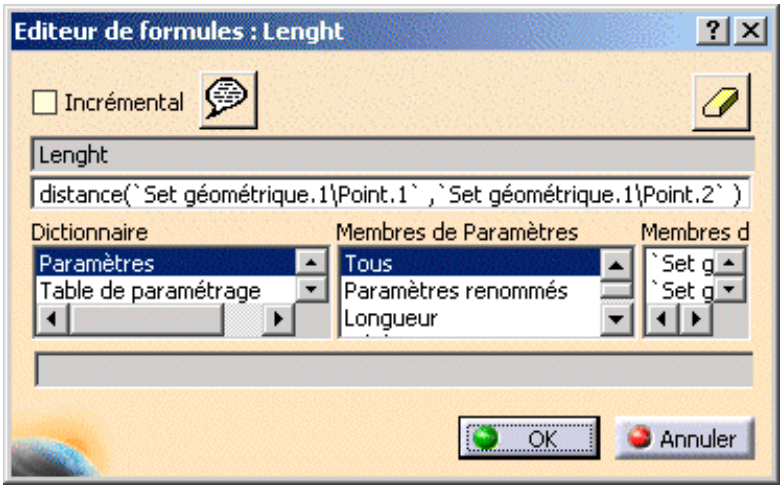
- Pour que le rapport de vérification soit créé correctement, sélectionnez **Outils->Options->Général->Paramètres et mesure ->Génération de rapport**, puis sélectionnez :

  - le fichier d'entrée XSL sous Fichier d'entrée XSL. (Un fichier XSL est indiqué par défaut. Cliquez ici pour obtenir une description du nouveau fichier XML.)

  - les paramètres que vous souhaitez voir apparaître dans le rapport sous Contenu du rapport.

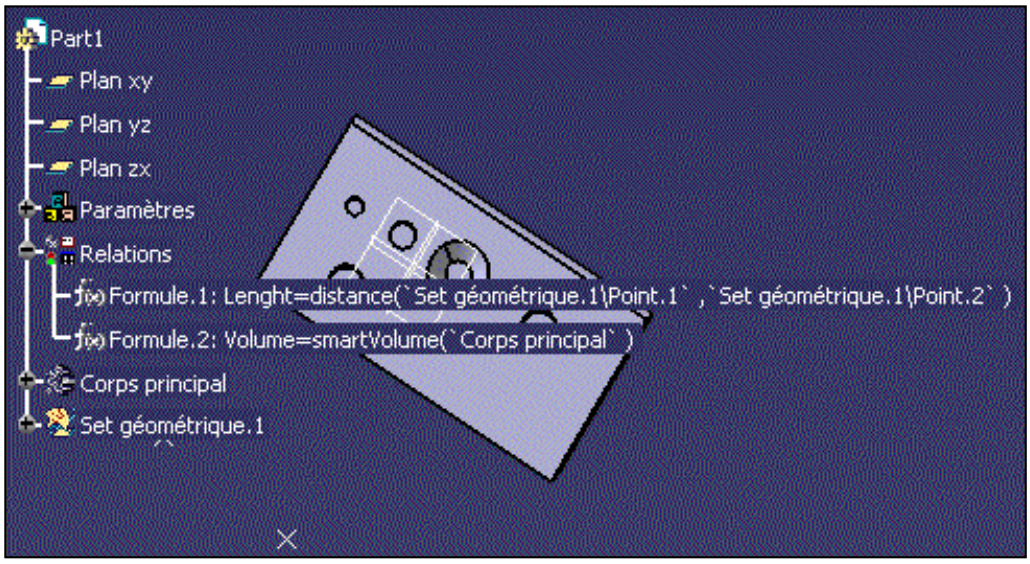
  - le répertoire de sortie sous Répertoire de sortie.

1. Ouvrez le fichier KwrCheckAnalysis.CATPart. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier **Knowledge Advisor**.
2. Créez un paramètre de type Longueur et affectez-lui une formule.
   - Cliquez sur l'icône f(x). L'éditeur de formules s'affiche.
   - Sélectionnez Longueur dans la liste déroulante afin de définir le type du paramètre, cliquez sur le bouton Créer paramètre de type, modifiez le nom du paramètre (dans cet exemple, Longueur), puis cliquez sur le bouton Ajouter formule. L'éditeur de formules s'affiche.
   - Sous Dictionnaire, sélectionnez Mesures, puis cliquez deux fois sur distance(Body,Body). Placez le curseur avant la virgule, puis cliquez deux fois sur Point.1 dans l'arbre de spécifications ou dans la géométrie. Placez le curseur avant la virgule, puis cliquez deux fois sur Point.2 dans l'arbre de



spécifications. Cliquez sur **OK**.

3. Créez un paramètre de type Volume et affectez-lui une formule.
   - Cliquez sur l'icône  . L'éditeur de formules s'affiche.
   - Sélectionnez Volume dans la liste déroulante afin de définir le type du paramètre, cliquez sur le bouton Créer paramètre de type, modifiez le nom du paramètre (dans cet exemple, Volume), puis cliquez sur le bouton **Ajouter formule**.
   - Sous Dictionnaire, sélectionnez Mesures de pièce, puis cliquez deux fois sur smartVolume. Placez le curseur entre les parenthèses et sélectionnez PartBody dans l'arbre de spécifications. Cliquez sur **OK**.

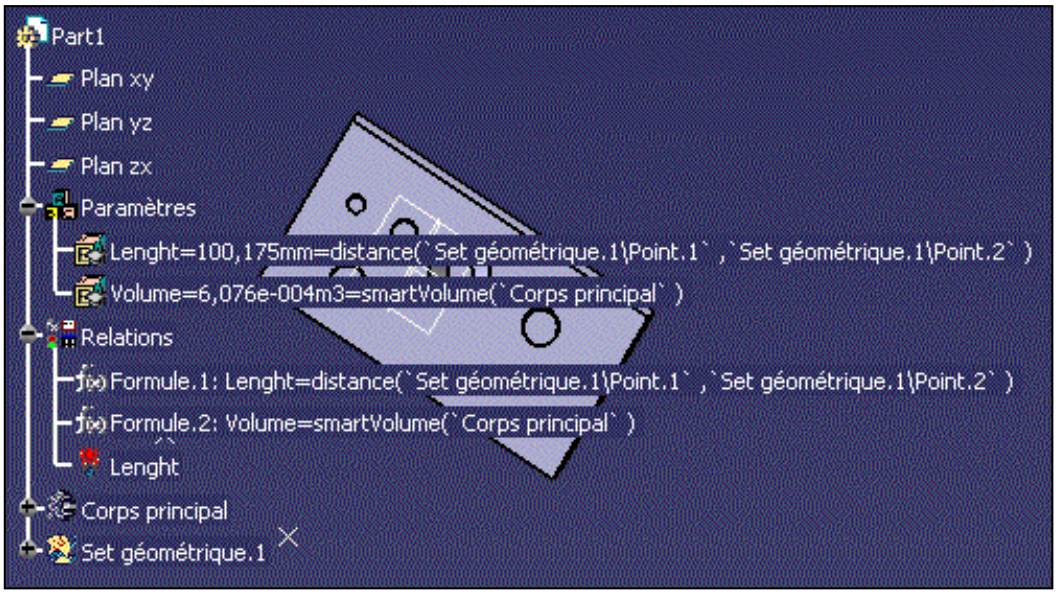


Les paramètres et les formules associées sont créés.

4. Ouvrez l'atelier Knowledge Advisor, cliquez sur l'icône **Vérification** (), modifiez le nom de la vérification (dans cet exemple, Longueur), puis cliquez sur **OK**. L'éditeur de vérifications s'affiche.
5. Dans l'éditeur, entrez le script suivant, puis cliquez sur **Appliquer** et sur **OK** .

```
Longueur > 150mm
```

La vérification Knowledge Advisor est créée.



6. Ouvrez l'atelier Knowledge Expert, puis cliquez sur l'icône Vérification et modifiez le nom de la vérification (dans cet exemple, HoleCheck). L'éditeur de règles Knowledge Expert s'ouvre.

7. Dans l'onglet Condition, entrez le script suivant :

	H:Hole
Editeur	H.Diameter > 15mm

8. Cliquez sur l'onglet Correction, sélectionnez VB Script dans la liste déroulante et entrez le script suivant dans l'éditeur :

```
Dim aHole as Hole
Set aHole = H.parent.Item(H.Name)
Dim diam As Length
Set diam = aHole.Diameter
diam.Value = 16
MsgBox("Correction performed on "&H.Name)
```

9. Dans le champ Commentaire pour la correction de l'onglet Correction, entrez la chaîne suivante, puis cliquez sur **OK** :

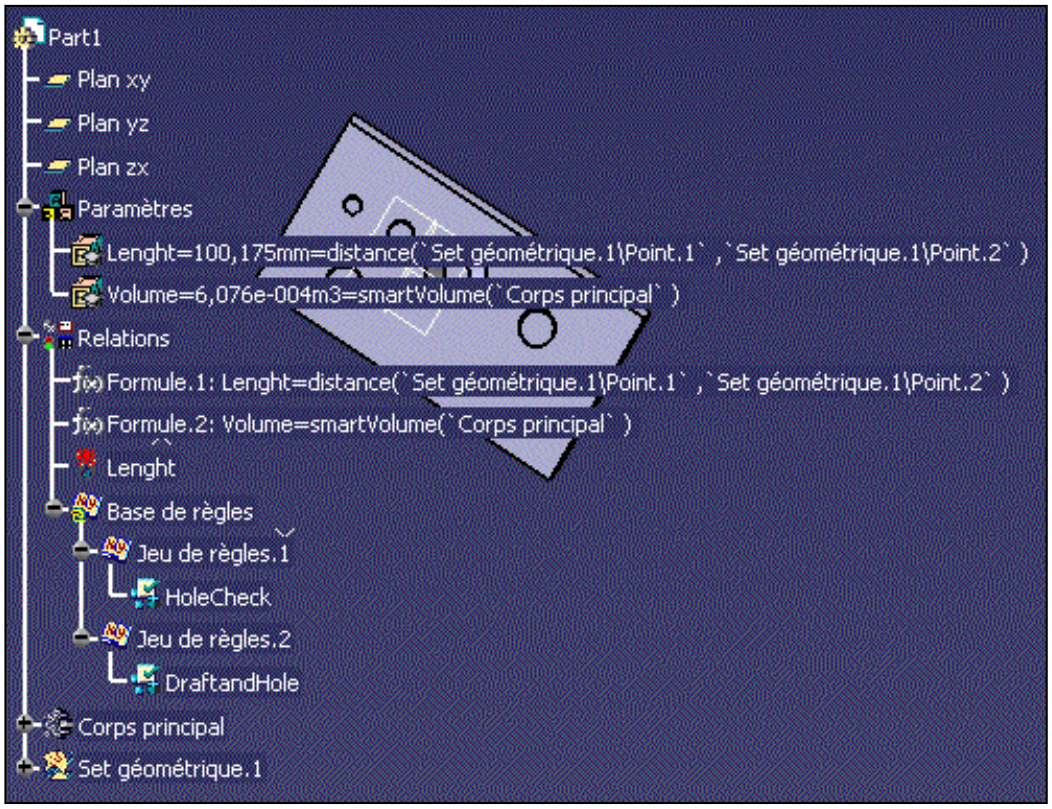
   Le diamètre des trous doit être supérieur à 15 mm.

10. Sous le noeud Relations, sélectionnez la base de règles et cliquez sur l'icône Vérification experte, puis modifiez le nom de la vérification (dans cet exemple, DraftandHole) et cliquez sur **OK**. L'éditeur de vérifications s'ouvre.

11. Dans l'onglet Condition, entrez le script suivant, puis cliquez sur **Appliquer** et sur **OK**.

	H:Hole ; D:Draft
Editeur	D.Activity AND H.Diameter > 12mm

Les vérifications sont créées (cliquez sur le graphique ci-contre afin de l'agrandir).



12. Cliquez sur l'icône dans la barre d'outils. L'outil d'analyse globale s'affiche.

13. Cliquez sur l'icône pour mettre à jour le statut des vérifications. Les voyants des vérifications deviennent rouges dans l'arbre de spécifications.

14. Cliquez sur l'icône. Une page xml s'ouvre ; elle contient les éléments ne vérifiant pas les conditions. Pour plus d'informations sur ce rapport, voir Personnalisation des rapports de vérification.

15. Cliquez sur l'icône pour exécuter la méthode de correction indiquée lors de la création de la vérification Knowledge Expert (reportez-vous à l'étape 9). Les vérifications ont été corrigées.

    Seule la vérification Knowledge Advisor (Longueur) ne peut pas être corrigée : la valeur du paramètre Longueur est 100.175 mm (comme indiqué dans le rapport). Or, elle doit être supérieure à 150 mm (comme indiqué dans le corps de la vérification).

16. Pour corriger la vérification, modifiez la valeur du paramètre Longueur.

    - Cliquez deux fois sur Point.1 dans la zone géométrique. La fenêtre Définition du point s'ouvre.
    - Dans le champ H:, remplacez la valeur du point par 150 mm. Cliquez sur **OK**. Le voyant de la vérification devient vert, indiquant que la vérification a abouti.

Utilisation de l'outil d'analyse des vérifications

L'outil d'analyse globale permet de gérer les vérifications Knowledge Expert et Knowledge Advisor, quel que soit leur emplacement dans l'arbre de spécifications. Il vous aide à comprendre le statut de validation de vos conceptions, permet de naviguer dans les vérifications et les violations, et met en surbrillance les composants non conformes. Vous pouvez :

- accéder aux informations sur les éléments non conformes ;
- regrouper des informations sur les objets et les vérifications ;
- exécuter des corrections automatiques, le cas échéant.

L'outil d'analyse globale est accessible au niveau de la session, en cliquant sur l'icône située dans la barre d'outils. Cette icône vous fournit un résultat des vérifications simple :

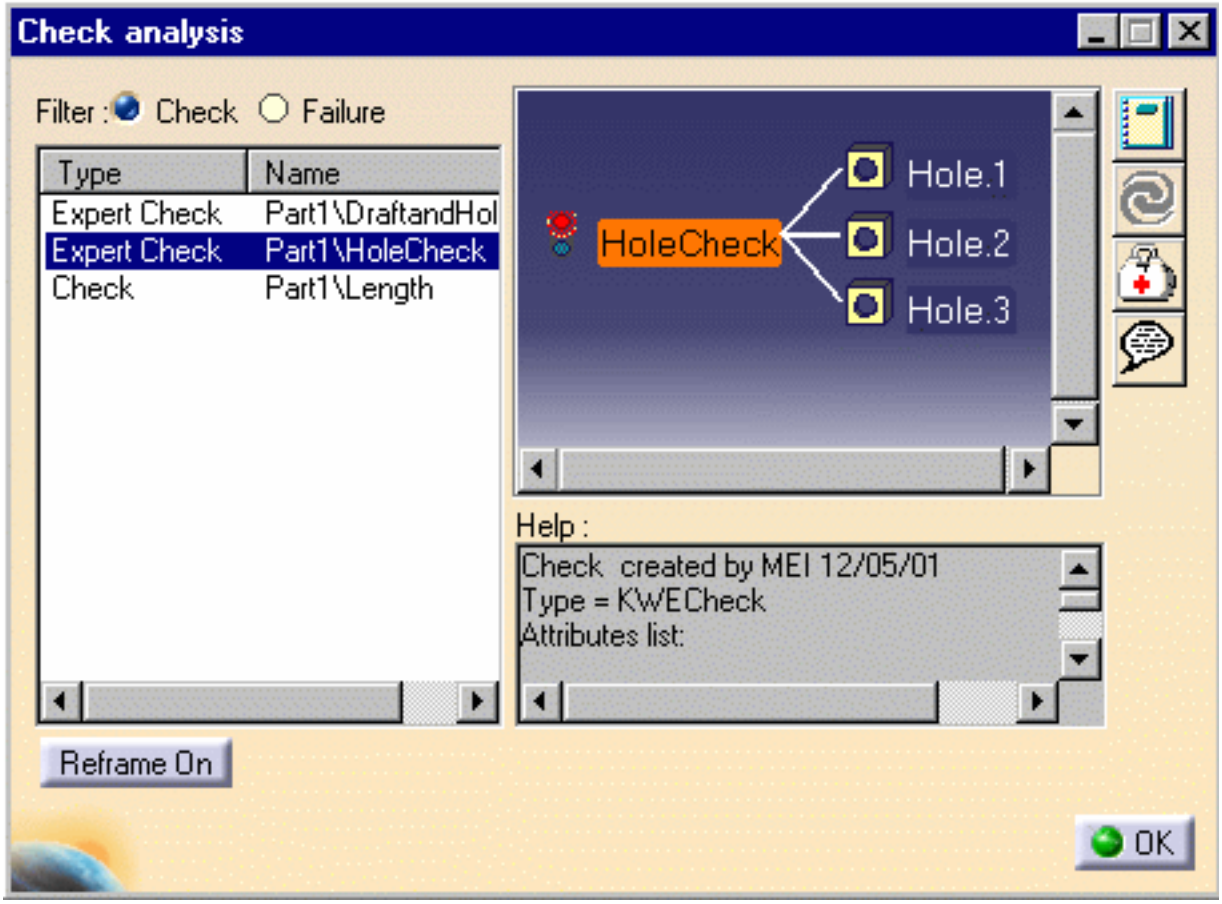
 Toutes les vérifications sont mises à jour et ont pu être déclenchées.

 Les vérifications doivent être mises à jour.

 Toutes les vérifications sont mises à jour et au moins une d'entre elles est incorrecte.

Fenêtre de l'outil Analyse des vérifications

Cliquez sur l'icône  dans la barre d'outils afin d'accéder à la fenêtre Analyse des vérifications.



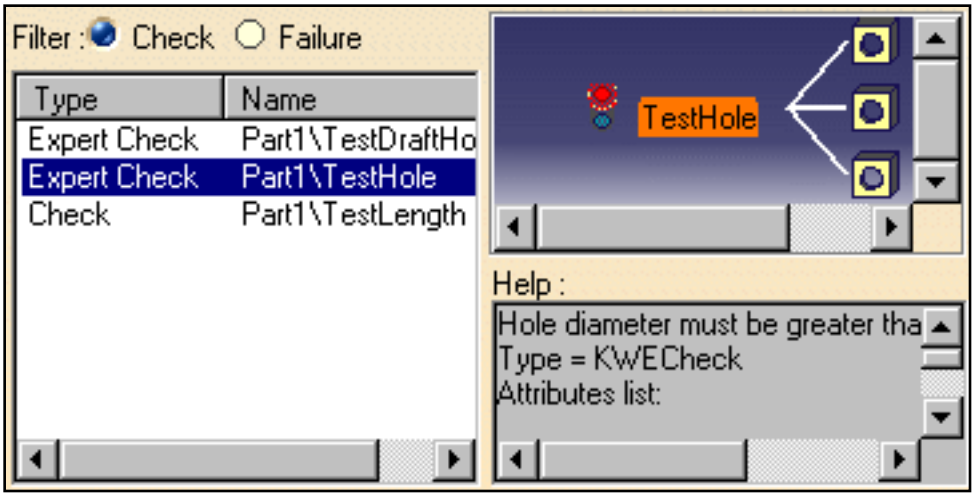
Section Filtre

Cette option vous permet d'appliquer un filtre aux vérifications ou aux éléments non conformes.

Vérification	Seules les vérifications Knowledge Expert et Knowledge Advisor ayant échoué lors de la mise à jour du rapport de vérification sont affichées.	Type / Name Expert Check Part1\TestDraftH Expert Check Part1\TestHole Check Part1\TestLength
Echec	Tous les éléments n'ayant pas satisfait les conditions de conformité lors de la mise à jour du rapport de vérification sont affichés.	Type / Name Simple hole Part1\Hole.1 Simple hole Part1\Hole.2 Tapered hole Part1\Hole.3 Length Part1\Part1\Leng

Section Aide

Pour afficher la section d'aide associée à chaque élément de la liste, cliquez deux fois sur l'élément souhaité. La fenêtre suivante s'affiche :



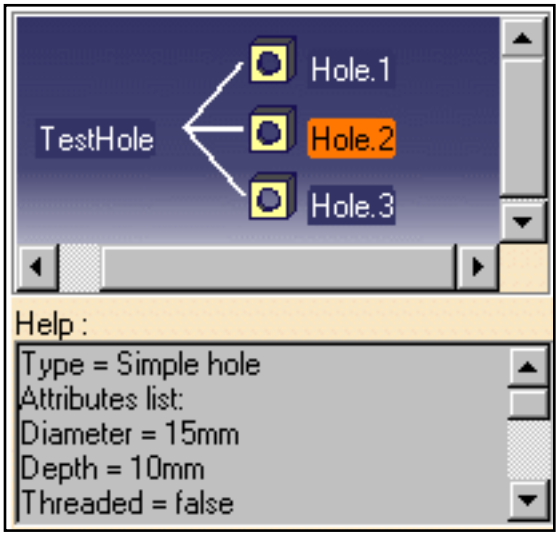
La vérification, ses éléments et leur statut courant sont affichés dans cette fenêtre.

Les éléments entrés lors de la création de la vérification s'affichent :

- Commentaires associés
- Type
- Attributs
- Variables
- Nom
- Propriétaire de la vérification...

Dans le graphique ci-dessus, la vérification sélectionnée est TestHole ; elle contrôle les trous du fichier CATPart (3 d'entre eux ne sont pas contrôlés car leur diamètre est inférieur à 15 mm). Les attributs s'affichent en fonction des données entrées lors de la création de la vérification.

Notez qu'il est également possible de sélectionner les éléments associés à la vérification.
Pour ce faire, cliquez deux fois sur l'élément souhaité dans la fenêtre : la section d'aide affiche les informations relatives à cet élément (reportez-vous au graphique ci-contre).



Barre d'outils

Cliquez sur cette icône pour créer le rapport de vérification personnalisable. Pour plus d'informations sur le rapport de vérification, voir Personnalisation des rapports de vérification.

Cliquez sur cette icône pour résoudre les vérifications créées dans votre document.

Cliquez sur cette icône pour exécuter la méthode de correction indiquée dans l'éditeur de vérifications lors de la création de la vérification. Pour obtenir un exemple, reportez-vous à la section Exécution d'une analyse globale des vérifications.

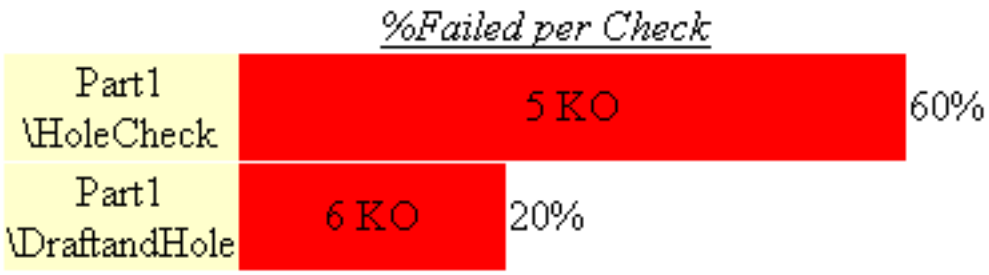


Cliquez ici pour afficher l'URL associée à l'objet ou pour en affecter une à un objet. Pour plus d'informations, reportez-vous à la section Association d'adresses URL et de commentaires à des paramètres ou à des relations.

Présentation du rapport de vérification par défaut

Le rapport de vérification par défaut présente les vérifications Knowledge Expert et Knowledge Advisor ayant échoué.

1. Part1\Length
2. Part1\HoleCheck
3. Part1\DraftandHole



Cette fenêtre répertorie les vérifications ayant échoué et présente, pour chaque vérification Knowledge Expert, le pourcentage des éléments ne vérifiant pas les conditions.

Rapport de vérification Knowledge Advisor

La fenêtre des vérifications Knowledge Advisor répertorie les vérifications Knowledge Advisor ayant échoué et affiche les éléments suivants :
- le corps de la vérification (ici, d'une longueur > à

Check advisor: Part1\Length

Body

```
/*Check created by MEI 11/29/01*/
Length > 150mm
```

This check operates on :

- *Part1\Length = 100.175mm*

150 mm) ;
- les éléments sur lesquels porte la vérification (ici, la formule de la longueur).

Rapport de vérification Knowledge Expert

Part1\HoleCheck

Comment					
Input	Part1 \Hole.1	Part1 \Hole.2	Part1 \Hole.3	Part1 \Hole.4	Part1 \Hole.5

✗	Part1\Hole.1
✗	Part1\Hole.2
✗	Part1\Hole.3
✓	Part1\Hole.4
✓	Part1\Hole.5

La fenêtre des vérifications Knowledge Expert répertorie les vérifications Knowledge Expert ayant échoué et affiche les éléments suivants :
- les éléments d'entrée soumis à la vérification ;
- les éléments ne vérifiant pas les conditions (ici, Hole.1, Hole.2 et Hole.3).

Notez que ce rapport ne doit pas être utilisé pour créer des macros ou d'autres fichiers. Il est fourni à titre d'information uniquement.

Personnalisation des rapports de vérification

Les rapports générés par l'éditeur d'analyse globale des vérifications peuvent être personnalisés.

Vous pouvez choisir d'afficher un rapport xml ou html.

Affichage d'un rapport HTML

Pour créer un rapport html pendant l'exécution de l'analyse des vérifications, sélectionnez **Outils->Options->Général->Paramètres et mesure**, puis sélectionnez l'onglet **Génération de rapport**. Sélectionnez **Html** dans la zone de sélection **Configuration du rapport sur les vérifications**.

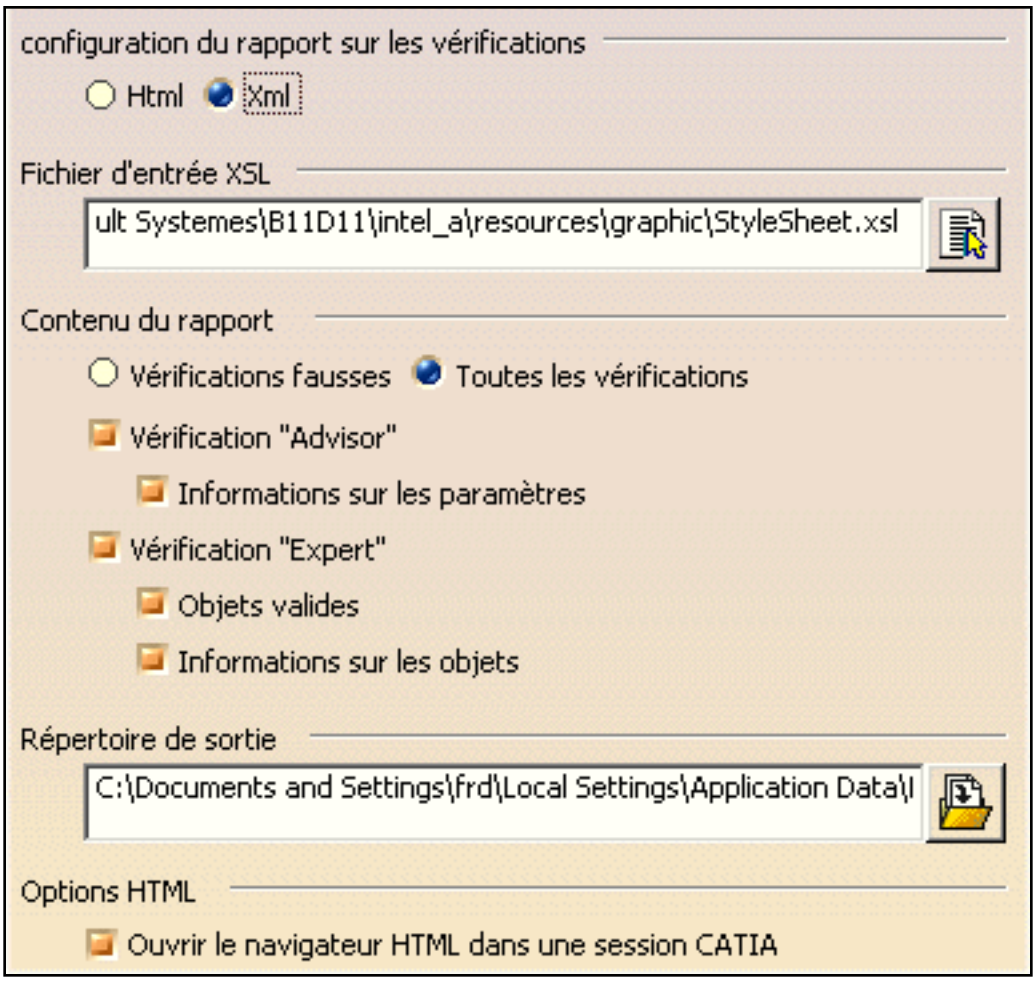
Dans le cas présent, seules les options **Vérification Knowledge Advisor**, **Vérification Knowledge Expert** et **Objets valides** sont disponibles dans la zone Contenu du rapport. Vous pouvez indiquer dans le champ **Sélectionner un répertoire de sortie** le répertoire de sortie contenant le nouveau rapport HTML.



Si vous utilisez un navigateur Netscape, sélectionnez **Html**.

Affichage d'un rapport XML

Pour afficher un rapport XML pendant l'exécution de l'analyse des vérifications, sélectionnez **Outils->Options->Général->Paramètres et mesure**, puis sélectionnez l'onglet **Génération de rapport**. Sélectionnez **Xml** dans la zone de sélection **Configuration du rapport sur les vérifications**. La fenêtre suivante s'ouvre :



L'onglet **Génération de rapport** se compose de quatre champs différents : Fichier d'entrée XSL, Contenu du rapport, Sélectionner un répertoire de sortie et Options HTML.

Champ Fichier d'entrée XSL

Ce champ vous permet de sélectionner la feuille de style XSL qui sera appliquée au nouveau rapport XML. Le fichier StyleSheet.xsl est le fichier XSL par défaut, mais vous pouvez utiliser votre modèle.

Champ Contenu du rapport

Vérifications fausses	Si cette option est activée, le nouveau rapport contiendra des informations sur les vérifications fausses uniquement.
Toutes les vérifications	Si cette option est activée, le nouveau rapport contiendra des informations sur toutes les vérifications contenues dans le document.
Vérification Knowledge Advisor	Si cette option est activée, le nouveau rapport contiendra des informations sur toutes les vérifications Knowledge Advisor contenues dans le document.

	Informations sur les paramètres	Si cette option est activée, le nouveau rapport contiendra des informations sur les paramètres des vérifications Knowledge Advisor.
Vérification Knowledge Expert		Si cette option est activée, le nouveau rapport contiendra des informations sur toutes les vérifications Knowledge Expert contenues dans le document.
	Objets valides	Si cette option est activée, le nouveau rapport contiendra des informations sur tous les objets valides après les vérifications Knowledge Expert, ainsi que des informations sur les paramètres de ces objets (diamètre, profondeur, pas de vis, ...).
	Informations sur les objets	Si cette option est activée, le nouveau rapport contiendra des informations sur tous les objets contenus dans les vérifications Knowledge Expert, ainsi que des informations sur les paramètres de ces objets (diamètre, profondeur, pas de vis, ...).

Champ Répertoire de sortie

Ce champ vous permet de sélectionner le répertoire de sortie contenant le nouveau rapport XML.

Champ Options HTML

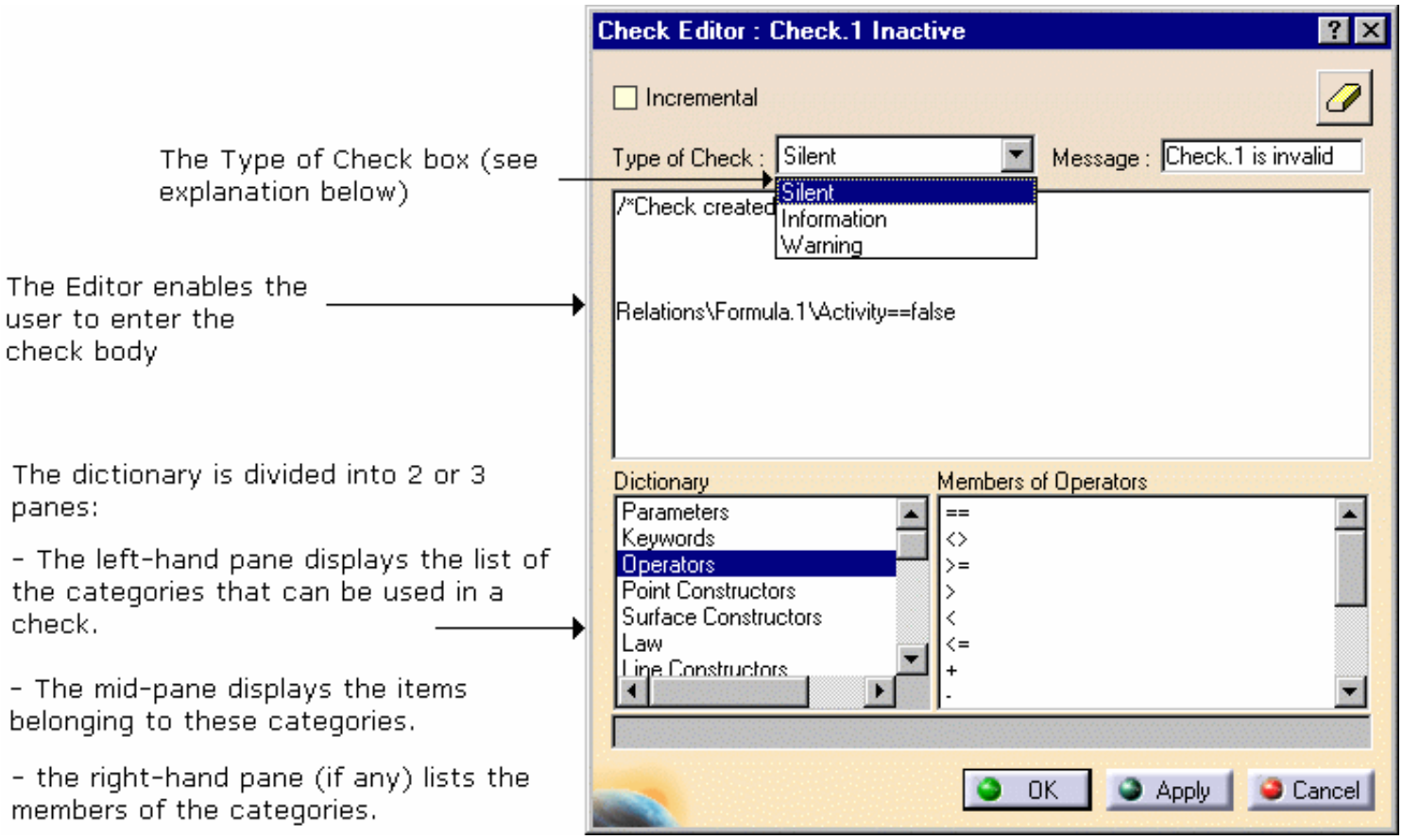
Cette option est disponible sous Windows uniquement. Elle vous permet de définir si le rapport sera ouvert lors d'une session *CATIA* (dans ce cas, la case doit être cochée), ou lors d'une session Internet Explorer (dans ce cas, la case ne doit pas être cochée).



Notez qu'il est fortement déconseillé d'utiliser ce rapport comme base pour des macros ou pour d'autres applications. Il est fourni à titre d'information uniquement.

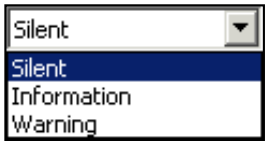
Utilisation de l'éditeur de vérifications

L'éditeur de vérifications est destiné à vous aider à entrer le corps de la vérification à l'aide du dictionnaire. Il se compose de trois zones différentes :



Il existe trois types de vérification différents :

- Vérifications de type Silent
- Vérifications de type Information
- Vérifications de type Warning



Selon le type de vérification choisi et le résultat de cette vérification, CATIA vous informera que :

	la vérification a réussi, ou que	la vérification a échoué.
Icône Relation dans l'arbre de spécifications		
Vérification de type Silent	pas de message affiché	pas de message affiché
Vérification de type **Information**	pas de message affiché	 Le message spécifié lors de la création de la vérification apparaît dans une fenêtre d'information.

Vérification de type Warning	pas de message affiché		Le message spécifié lors de la création de la vérification apparaît dans une fenêtre d'avertissement.

Dans l'éditeur de vérifications, vous pouvez :

- Limiter la liste des paramètres affichés dans le dictionnaire. Il suffit de cliquer dans l'arbre de spécifications sur l'entité dont vous souhaitez afficher les paramètres. Si la case 'Incrémental' est cochée, seuls les paramètres du premier niveau sont affichés juste en dessous de l'entité sélectionnée. Sinon tous les paramètres de tous les niveaux apparaissent.
- Insérer la définition de l'entité dans une vérification. Pour cela, affichez l'arbre de spécifications et cliquez deux fois sur l'entité dont vous souhaitez insérer la définition.
- Vérifier si la syntaxe de la vérification est correcte. Pour cela, cliquez sur **Appliquer**.
- Effacer le contenu de la fenêtre d'édition. Pour cela, cliquez sur l'icône  .
- Ajouter la vérification au document. Pour cela, cliquez sur **OK**.



Pour plus d'informations sur le dictionnaire, voir Utilisation du dictionnaire. Pour plus d'informations sur les vérifications, voir Création d'une vérification.

Utilisation de la fonction Réaction



Cliquez sur l'icône Réactions pour créer un script capable de modifier certains attributs des composants lorsqu'un événement survient.

Une réaction est une entité qui réagit aux événements agissant sur sa ou ses sources en déclenchant une action. Elle permet de gérer les restrictions liées aux règles et aux comportements, et de créer des conceptions plus associatives et plus réactives.

Une réaction est une entité qui réagit aux événements.

La source peut être :

- une entité sélectionnée (ou une liste d'entités) ;
- un paramètre (le résultat d'un test).

Les événements peuvent être :

- des événements concernant les objets (création, suppression, mise à jour, glisser-déposer, modification des attributs) et les modifications de valeurs des paramètres ;
- Insérer/Remplacer le composant
- Glisser-Déposer

Une réaction est similaire à une règle dans le sens où :

- elle est stockée dans le modèle ;
- elle réagit aux modifications et peut les déclencher ;
- elle référence également d'autres objets et paramètres dans le document, et prend en charge le mécanisme de remplacement.

Cependant

- les entités de réaction permettent un meilleur contrôle lorsque l'action doit être déclenchée.
- les réactions permettent à l'utilisateur d'exécuter des actions plus complexes ; étant donné que vous avez un meilleur contrôle lors du déclenchement de l'action, et que vous n'êtes pas soumis aux limitations du mécanisme de mise à jour, vous pouvez utiliser toute la puissance de n'importe quelle API Visual Basic (dans *CATIA*, mais aussi dans d'autres applications d'automatisation...), et vous pouvez appeler une macro Visual Basic avec des arguments à partir d'une action.

- les réactions peuvent être écrites afin de personnaliser le mécanisme de mise à jour (pour optimiser les gabarits, par exemple) ;
- Elles peuvent réagir aux actions de l'utilisateur (instanciation d'un gabarit), à l'insertion d'un composant dans un assemblage, à la modification d'un paramètre...
- elles peuvent être stockées dans le modèle et peuvent être intégrées à la définition d'une copie optimisée ou d'un gabarit.



L'entité Réaction n'est pas intégrée à la mise à jour, contrairement à la règle. Si une réaction est déclenchée lors de la mise à jour du document, cela peut avoir une incidence sur ce dernier.

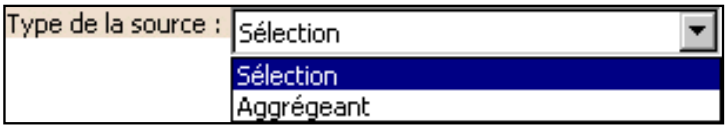
Utilisation de la fenêtre Réaction
Création d'une réaction : événement DragAndDropEvent
Création d'une réaction : événement AttributeModification
Création d'une réaction : événement Insert
Création d'une réaction : événement Inserted
Création d'une réaction : événement Remove
Création d'une réaction : événement BeforeUpdate
Création d'une réaction : événement ValueChange
Utilisation d'une réaction avec un gabarit :événement Instantiation
Utilisation d'une réaction Knowledge Advisor avec un modèle de document : événement Instantiation
Création d'une réaction : événement Update
Création d'une réaction : événement File Content Modification
Création d'une boucle dans une réaction

Utilisation de la fenêtre de la fonction Réaction

Vous pouvez accéder à la fenêtre Réaction en cliquant sur l'icône  dans l'atelier Knowledge Advisor.

La fenêtre Réaction se compose de trois champs principaux : Type de la source, Source et Action.

Champ Type de la source



- L'option **Sélection** permet de sélectionner manuellement un ou plusieurs éléments dans l'arbre de spécifications ou dans la géométrie. Ces éléments s'afficheront dans le champ **Sources**.
- L'option **Agrégeant** vous permet de lier l'action à une entité de la géométrie ou de l'arbre de spécifications (voir Utilisation de la fonction Réaction de Knowledge Advisor : événement DragAndDrop) lorsque l'entité réaction est liée à un trou, par exemple.
  Pour lier la réaction à un objet de la géométrie, cliquez sur le champ **Destination** et sélectionnez un objet dans l'arbre de spécifications ou dans la géométrie.

Champ Sources

Une réaction est une entité qui réagit aux événements (reportez-vous au paragraphe Evénements disponibles ci-dessous) sur un objet qui est appelé "source" et qui déclenche une action.

Le champ Sources affiche les éléments sélectionnés auxquels la réaction sera liée (ceci est possible uniquement si le type de source **Sélection** est sélectionné).

Evénements disponibles

Les événements disponibles contenus dans cette liste déroulante dépendent du type de source sélectionné dans le champ **Type de la source**. La réaction sera déclenchée lorsque l'un des événements détaillés ci-dessous se produira.

Evénements disponibles	Explications
AttributeModification	La réaction est déclenchée par la modification de l'état d'un attribut. Ceci est possible uniquement si l'option **Sélection** est sélectionnée.
BeforeUpdate	La réaction est déclenchée avant la mise à jour d'une entité.
DragAndDrop	La réaction est déclenchée après le glisser-déposer d'une entité.
Insert	La réaction est déclenchée lorsqu'une entité est insérée.
Inserted	La réaction est déclenchée après l'insertion d'une entité.
Instantiation	La réaction est déclenchée lorsqu'un gabarit est instancié.
Remove	La réaction est déclenchée lorsqu'une entité est supprimée.
Update	La réaction est déclenchée juste après la mise à jour d'une entité.

ValueChange	La réaction est déclenchée par la modification de la valeur d'un paramètre. Ceci est possible uniquement si l'option **Sélection** est sélectionnée.
FileContentModification	La réaction est déclenchée chaque fois que le fichier associé à la table de paramétrage est modifié.

Champ Action

L'action est déclenchée par une réaction résultant d'événements sur un objet. Ce champ permet de sélectionner le langage dans lequel vous souhaitez écrire et éditer l'action (VB ou le langage Knowledge Advisor).

Bouton Editer l'action



Cliquez sur ce bouton pour accéder à l'éditeur d'action.

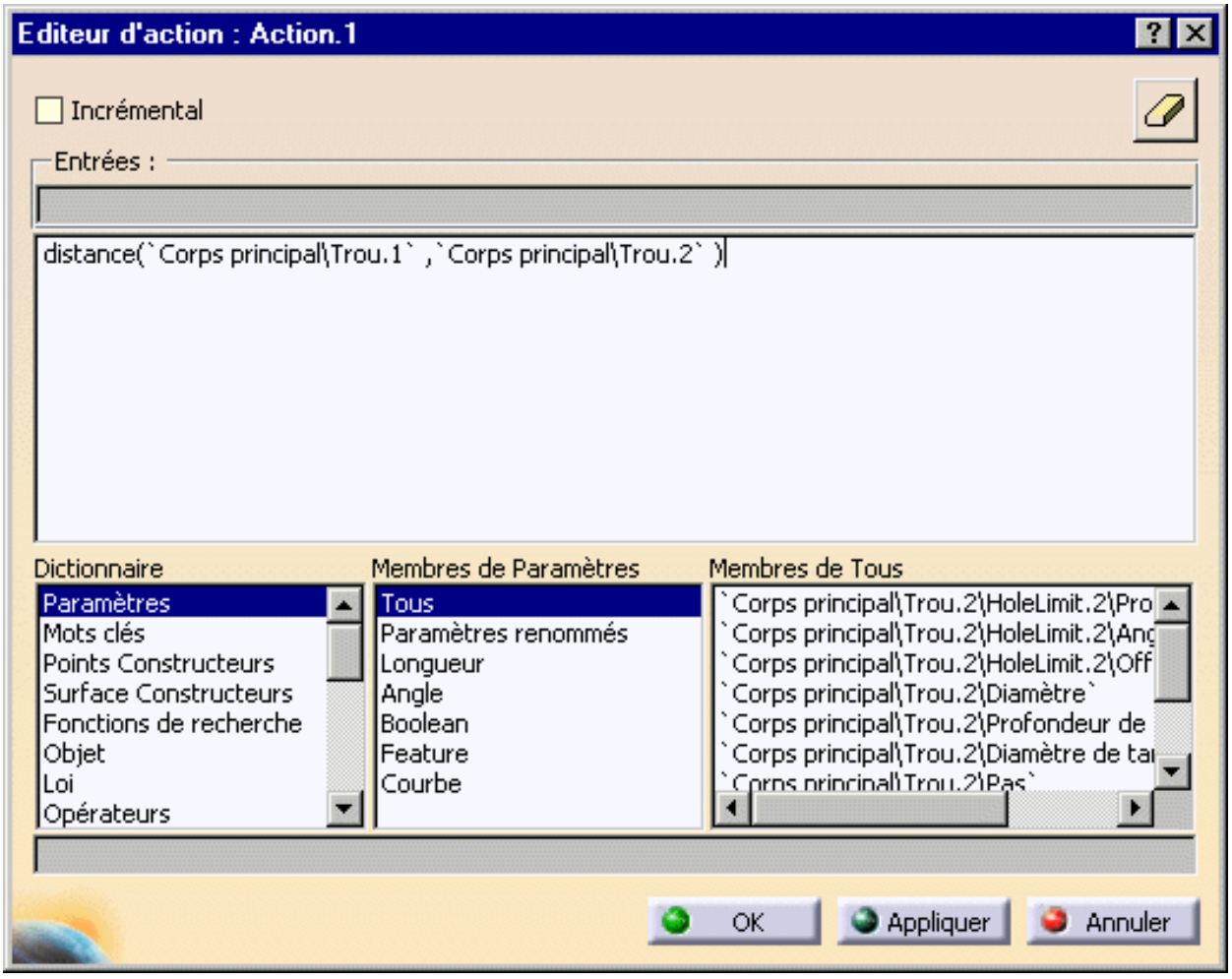
Editeur d'action

L'éditeur d'action qui s'affiche dépend du langage sélectionné dans le champ **Action**.

Si l'option **Action** est sélectionnée, la fenêtre suivante s'affiche :

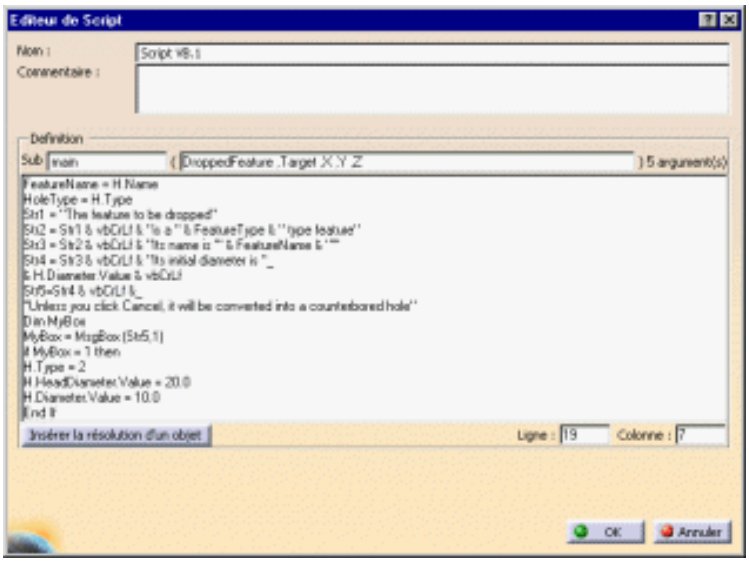
La sous-fenêtre d'édition permet d'entrer le corps de l'action.

Le **Dictionnaire** se divise en deux ou trois sous-fenêtres, selon la catégorie sélectionnée :
- La sous-fenêtre de gauche contient les catégories pouvant être utilisées pour une action.
- La sous-fenêtre du milieu répertorie les objets appartenant à la catégorie



sélectionnée.
- La sous-fenêtre de droite contient les membres de la catégorie sélectionnée.

Si l'option **Action VB** est sélectionnée, la fenêtre suivante s'affiche :



- Le champ **Nom** permet d'entrer un nom de script VB.

- Le champ **Commentaire** permet d'entrer un commentaire associé au script VB.

- Le champ **Editeur** permet d'entrer le script VB.

- Le bouton **Insérer la résolution d'un objet** permet de sélectionner un objet dans l'arbre de spécifications ou dans la géométrie et d'ajouter automatiquement sa résolution au script.

(Cliquez sur le graphique ci-contre afin de l'agrandir.)

Création d'une réaction : événement DragAndDrop
Création d'une réaction : événement Insert
Création d'une réaction : événement Inserted
Création d'une réaction : événement Remove

Création d'une réaction : événement AttributeModification
Création d'une réaction : événement BeforeUpdate
Création d'une réaction : événement ValueChange
Utilisation d'une réaction avec un gabarit : événement Instantiation
Utilisation d'une réaction avec un modèle de document : événement Instantiation
Création d'une réaction : événement Update
Création d'une réaction : événement FileContentModification

Création d'une réaction : événement DragAndDrop

Dans cette tâche, vous apprendrez à utiliser l'événement DragAndDrop dans la fonction Réaction. Dans l'exemple ci-dessous, vous faites un glisser-déposer avec un trou, ce qui déclenche une réaction.

a) Lorsque la règle est déclenchée, les informations suivantes s'affichent dans une fenêtre VB :

- le type d'entité ;
- son nom et son diamètre initial.

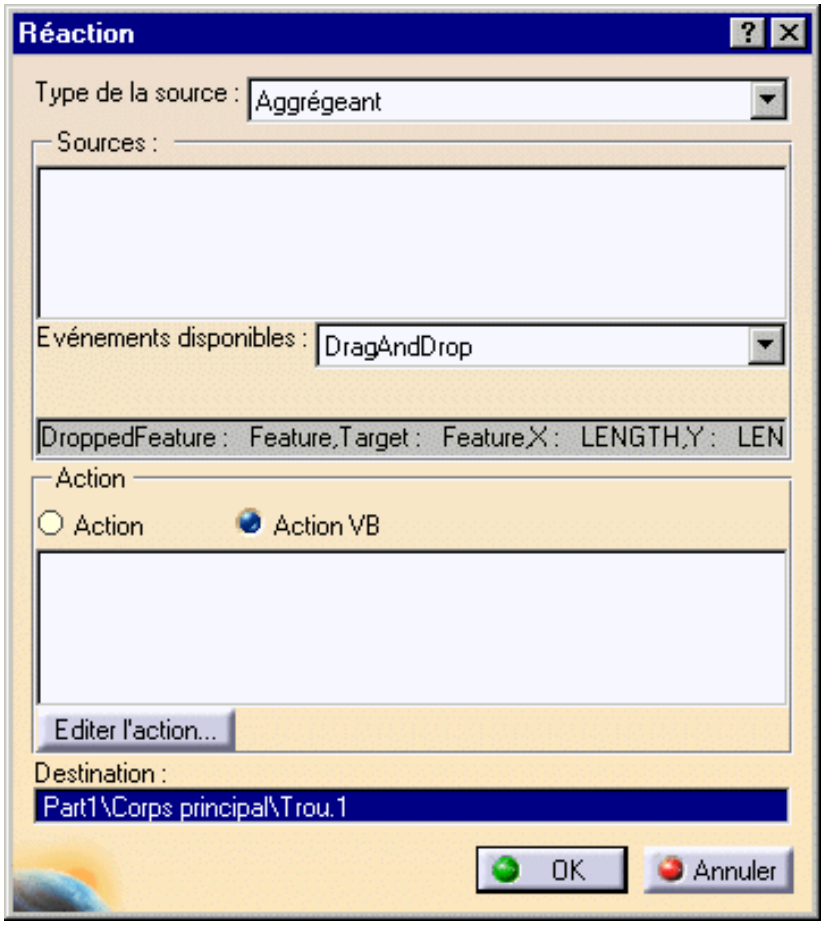
b) Vous êtes invité à cliquer sur **OK** pour convertir le trou en trou alésé ou sur **Annuler** pour ignorer la conversion.

Notez que cette tâche pouvait être exécutée dans le passé à l'aide de la fonction Comportement remplacée désormais par la fonction Réaction.

Pour plus d'informations sur les caractéristiques de la fonction Réaction, reportez-vous à la section Utilisation de la fonction Réaction.

1. Créez un fichier .CATPart ainsi qu'une extrusion avec un trou ou ouvrez le fichier KwrReactionPad.CATPart.
2. Ouvrez l'atelier Knowledge Advisor et cliquez sur l'icône Réaction () Pour créer une réaction. La boîte de dialogue Réaction s'affiche :

- Dans le champ **Type de la source**, sélectionnez **Agrégeant** pour que la réaction s'applique au trou sélectionné dans la zone **Destination** (voir ci-dessous).

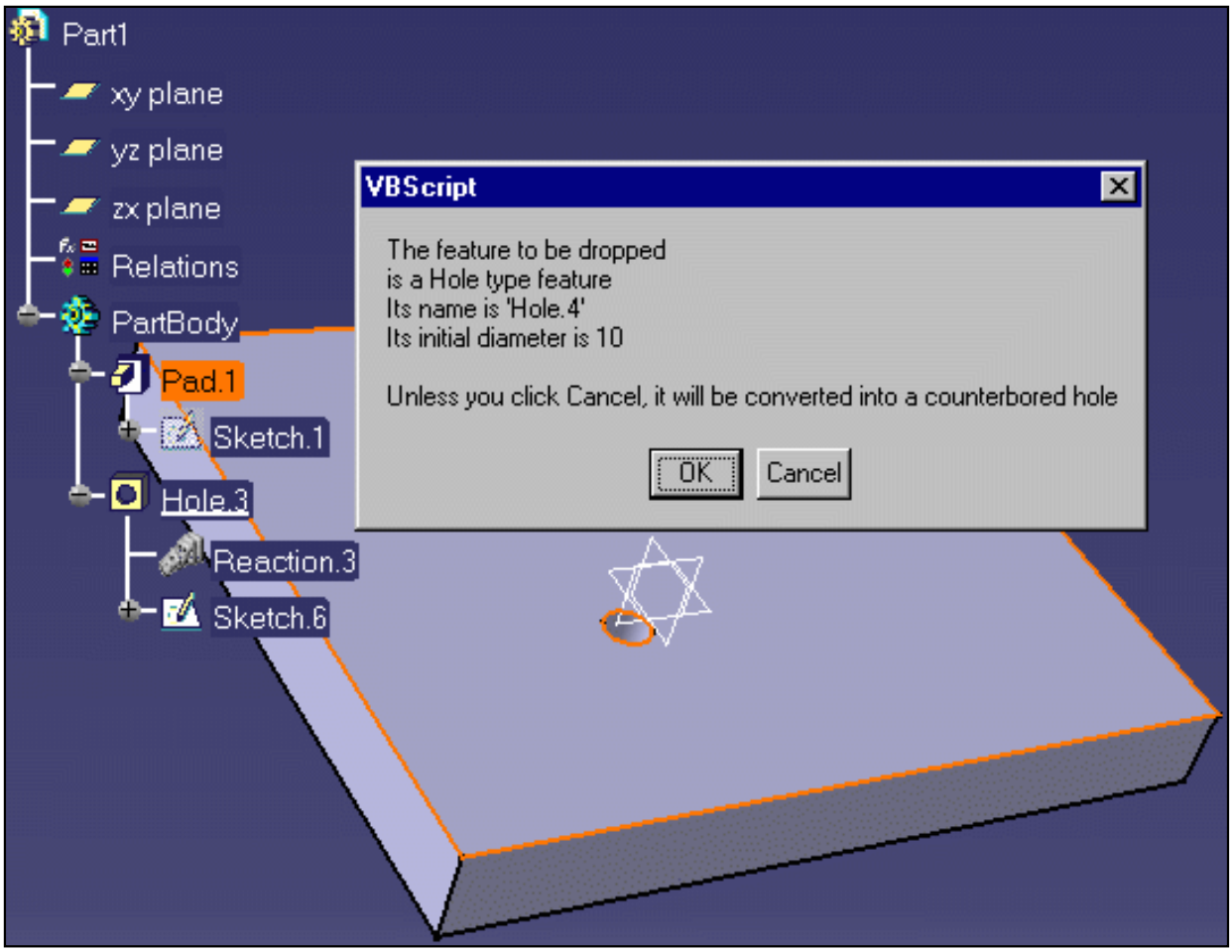


- Dans la liste **Evénements disponibles**, sélectionnez **Glisser-Déposer** pour que la réaction se produise lorsque le champ est glissé et déposé.
- Dans le champ **Action**, sélectionnez **Action VB** pour écrire l'action en Visual Basic (VB).
- Dans la boîte de dialogue Réaction, cliquez dans la zone **Destination** et sélectionnez Trou.1 dans l'arbre de spécifications.

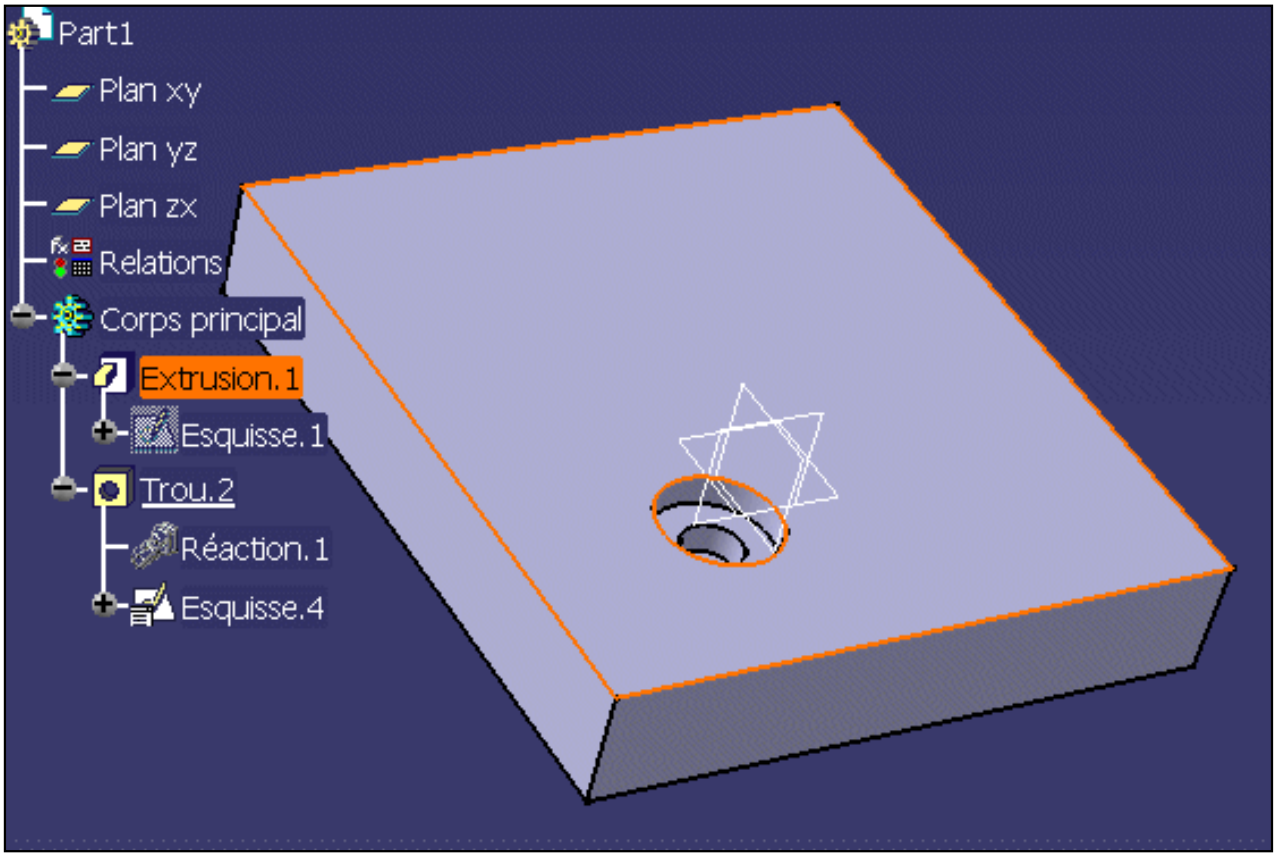
3. Cliquez sur le bouton **Editer l'action...**, collez le script suivant dans l'éditeur, puis cliquez deux fois sur **OK** :

```
Set H = DroppedFeature.Parent.Item(DroppedFeature.Name)
Dim FeatureType, FeatureName, HoleType
FeatureType = TypeName(H)
FeatureName = H.Name
HoleType = H.Type
Str1 = "Entité_à_déplacer"
Str2 = Str1 & vbCrLf & "est une entité de type " & FeatureType & ""
Str3 = Str2 & vbCrLf & "Son nom est '" & FeatureName & "'"
Str4 = Str3 & vbCrLf & "Son diamètre initial est "_
& H.Diameter.Value & vbCrLf
Str5=Str4 & vbCrLf &_
"Il sera transformé en trou alésé si vous ne cliquez pas sur Annuler."
Dim MyBox
MyBox = MsgBox (Str5,1)
if MyBox = 1 then
H.Type = 2
H.HeadDiameter.Value = 20.0
H.Diameter.Value = 10.0
End If
```

4. Ouvrez l'atelier Part Design. Dans la géométrie, sélectionnez le trou, puis faites un glisser-déposer avec lui. La boîte de dialogue suivante s'affiche :



5. Cliquez sur **OK**. Le trou est transformé en trou alésé (reportez-vois au graphique ci-dessous).



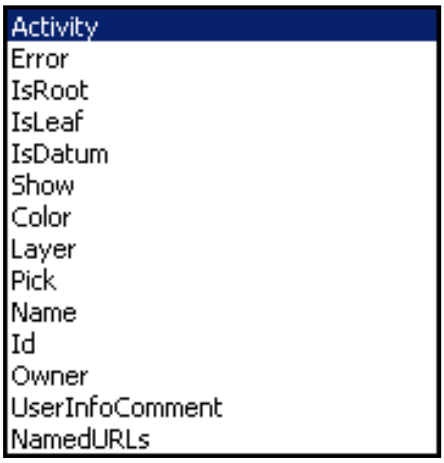
Pour plus d'informations sur la fenêtre de la fonction Réaction, voir Utilisation de la fenêtre de la fonction Réaction.

Création d'une réaction : événement AttributeModification

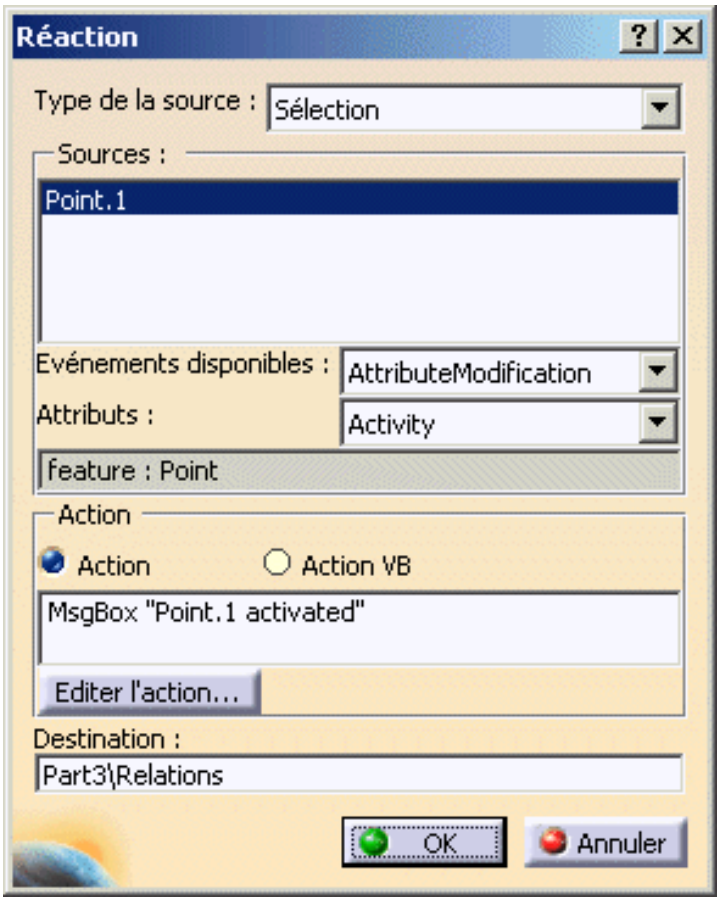
Dans cette tâche, vous apprendrez à utiliser l'événement AttributeModification dans la fonction Réaction. Dans l'exemple ci-dessous, vous allez créer une réaction basée sur l'activité (activation ou désactivation) d'un point. Si le point est activé, un message s'affiche.

Cette réaction est conçue pour réagir aux modifications d'attributs (couleur, état, nom, épaisseur...).

Notez que les événements de modification d'un attribut ne sont pas disponibles pour toutes les entités. Si cela s'avère être le cas pour l'entité que vous avez sélectionnée, sélectionnez l'attribut dans la fenêtre **Sélection de la source...** et utilisez l'événement ValueChange. La réaction sera déclenchée lors de la modification de l'attribut.



1. Ouvrez le fichier KwrAttributeModificationEvent.CATPart. Ce fichier contient le point désactivé.
2. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier **Knowledge Advisor** et cliquez sur l'icône **Réaction** () pour créer une réaction. La boîte de dialogue Réaction s'affiche :
   - Dans le champ **Type de la source**, sélectionnez **Sélection** ainsi que l'élément Point.1 dans l'arbre de spécifications.
   - Dans la liste déroulante **Evénements disponibles**, sélectionnez **AttributeModification**.
   - Dans la liste déroulante **Attributs**, sélectionnez **Activité** pour la réaction à lancer lors de l'activation ou de la désactivation du point.



- Dans le champ **Action**, sélectionnez **Action VB** et entrez le texte suivant dans l'éditeur : **MsgBox "Point.1 activé"**.
- Lorsque vous avez terminé, cliquez sur **OK**. La réaction créée s'affiche sous le noeud Relations dans l'arbre de spécifications.

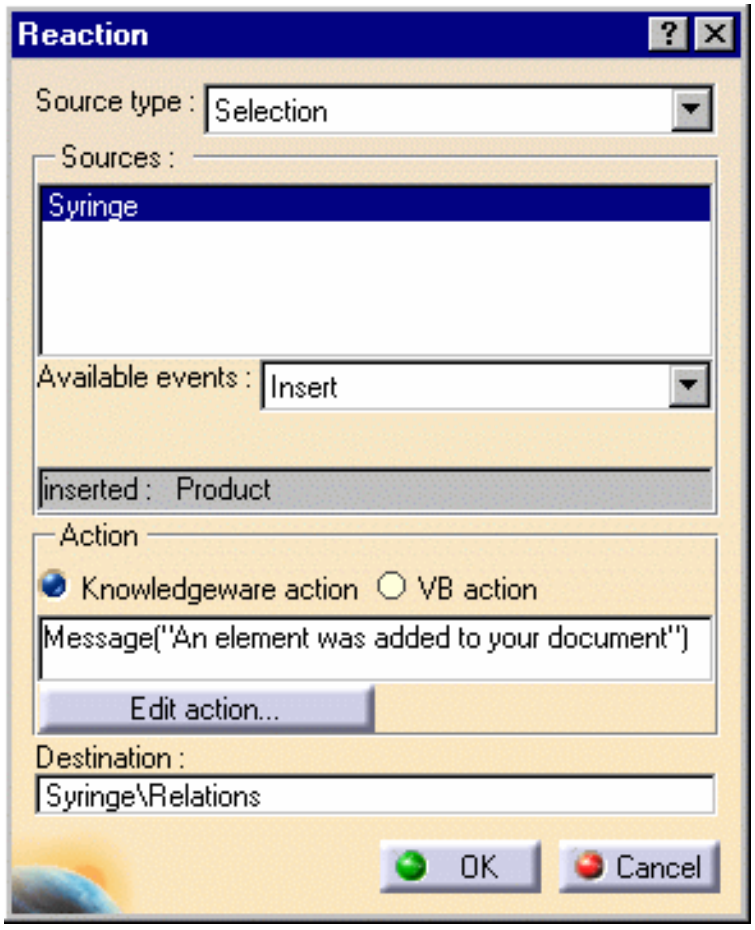
3. Cliquez deux fois sur Point.1 dans la géométrie pour ouvrir l'atelier Generative Shape Design.
4. Cliquez deux fois sur Point.1 et sélectionnez la commande **Objet Point.1->Activer**. Une boîte de dialogue indique que Point.1 est activé.
5. Cliquez sur **OK**. Point.1 est activé.

Pour plus d'informations sur la fenêtre de la fonction Réaction, voir Utilisation de la fenêtre de la fonction Réaction.

Création d'une réaction : événement Insert

Dans cette tâche, vous apprendrez à utiliser l'événement Insert dans la fonction Réaction. Dans l'exemple ci-dessous, vous allez insérer un élément dans le document CATProduct qui contient un message.

1. Ouvrez le fichier KwrSyringeAssembly.CATProduct.
2. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier Knowledge Advisor et cliquez sur l'icône Réaction () afin de créer une réaction. La boîte de dialogue Réaction s'affiche :



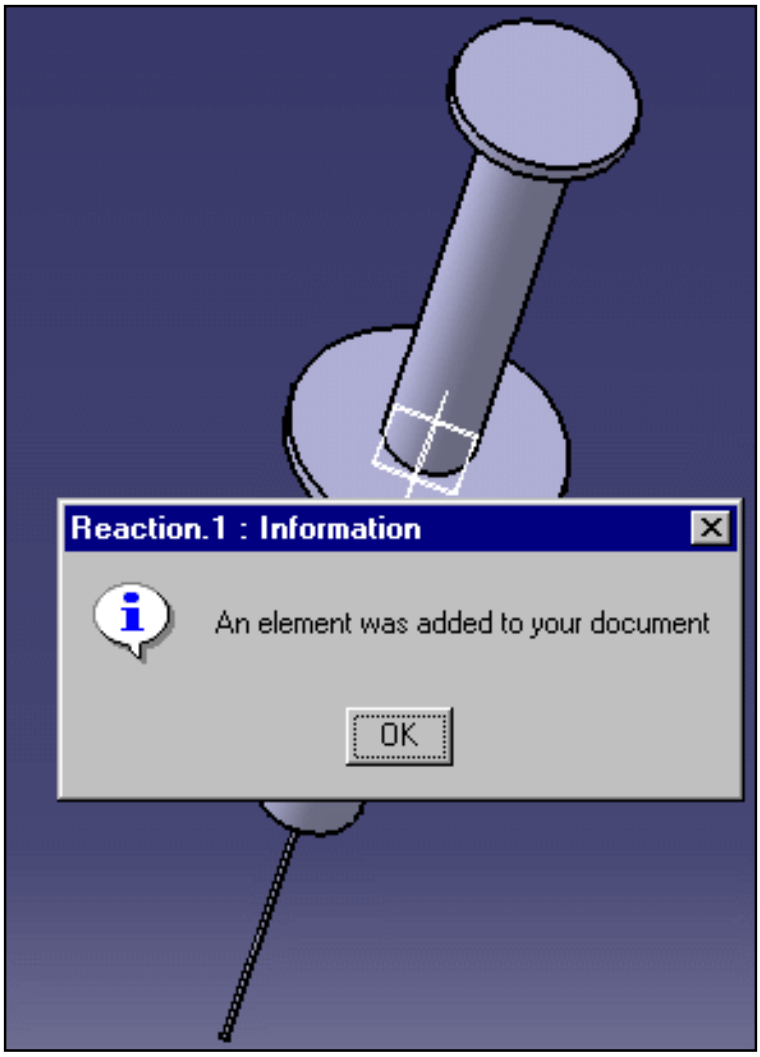
- Dans le champ **Type de la source**, sélectionnez **Sélection** pour que la réaction s'applique à l'élément sélectionné puis, dans l'arbre de spécifications, sélectionnez la seringue.
- Dans la liste **Evénements disponibles**, sélectionnez **Insert** pour que la réaction se produise lorsqu'un élément est inséré dans le document

CATProduct.

- Dans le champ **Action**, sélectionnez **Action Knowledgeware** et entrez le message suivant : Message("Un élément a été ajouté au document"). Ce message s'affichera chaque fois que vous insérerez un nouveau composant dans le document CATProduct.

- Cliquez sur **OK**. Une réaction est ajoutée au noeud Relations dans l'arbre de spécifications.

3. Cliquez deux fois sur la racine de l'arbre de spécifications, sélectionnez la commande **Insertion->Composant existant...** puis cliquez à nouveau sur la racine de l'arbre de spécifications. La boîte de dialogue Sélection de fichier s'affiche :
4. Sélectionnez le fichier KwrSyringePiston.CATPart et cliquez sur **Ouvrir**.
5. Le nouvel élément est inséré et la réaction est déclenchée. Le message suivant s'affiche :



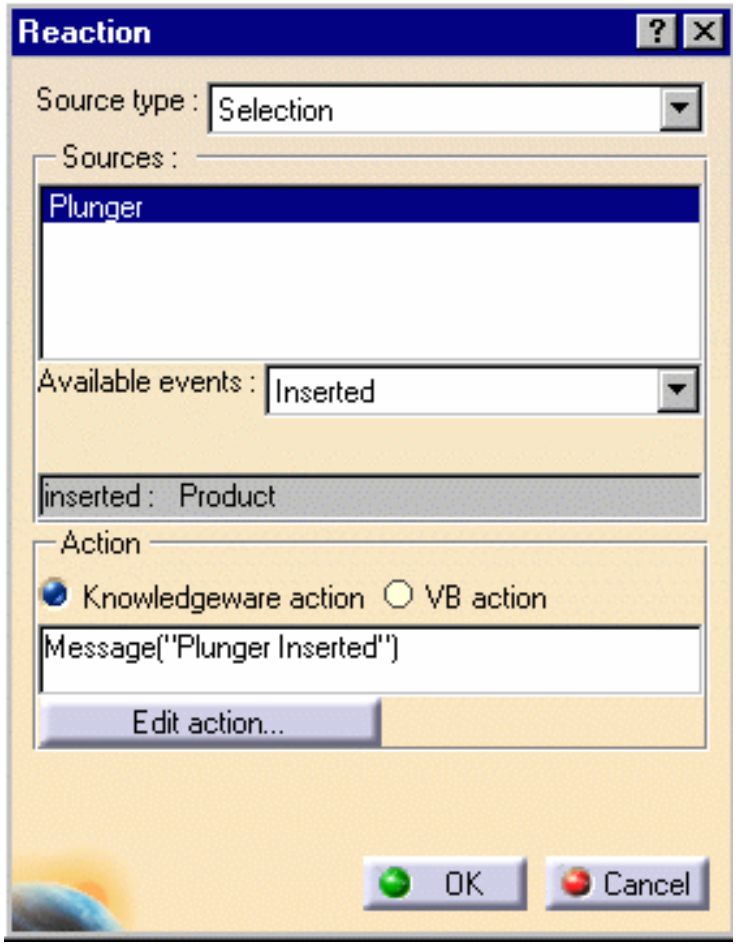
Pour plus d'informations sur la fenêtre de la fonction Réaction, voir Utilisation de la fenêtre de la fonction Réaction.

Création d'une réaction : événement Inserted

Dans cette tâche, vous apprendrez à utiliser l'événement Inserted dans la fonction Réaction. Dans l'exemple ci-dessous, vous allez insérer un élément dans le document CATProduct qui contient un message.

1. Créez un fichier CATProduct nommé Container.CATProduct et insérez le fichier KwrSyringeContainer.CATPart à l'aide de la commande **Insertion->Composant existant...**. Enregistrez le fichier, puis fermez-le.
2. Créez un fichier CATProduct nommé Plunger.CATProduct, renommez la racine de l'arbre de spécifications Plunger puis insérez le fichier KwrSyringePiston.CATPart à l'aide de la commande **Insertion->Composant existant...**. Fermez le fichier.
3. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier Knowledge Advisor et cliquez sur l'icône Réaction () afin de créer une réaction. La boîte de dialogue Réaction s'affiche :

   - Dans le champ **Type de la source**, sélectionnez **Sélection** pour que la réaction s'applique à l'élément sélectionné (dans cet exemple, Plunger).

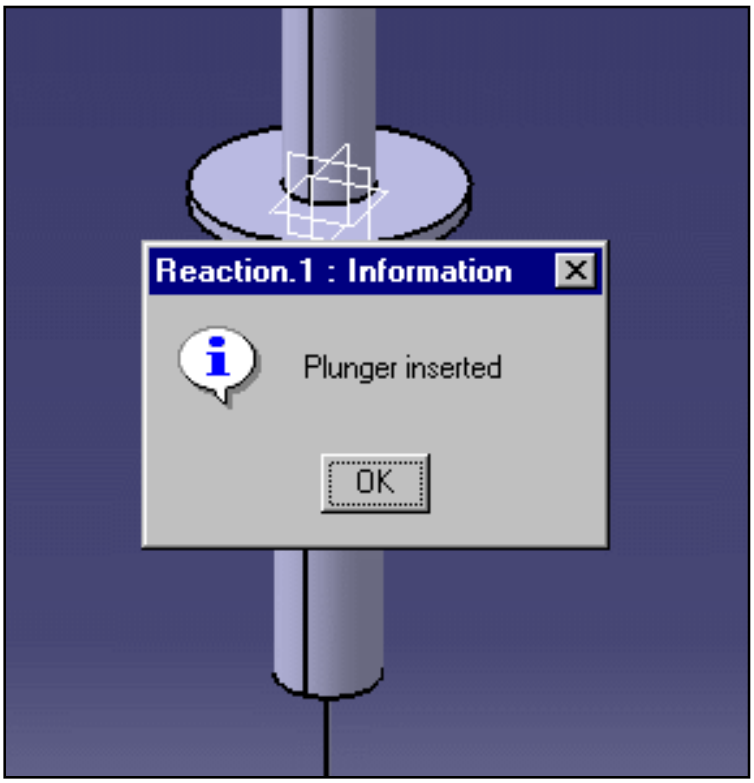


- Dans la liste **Evénements disponibles**, sélectionnez **Inserted** pour que la réaction soit lancée lorsque l'élément Plunger est inséré dans le document CATProduct.
- Dans le champ **Action**, sélectionnez **Action Knowledgeware** et entrez le message suivant : Message("Plunger inséré"). Ce message s'affichera chaque fois que l'élément Plunger sera inséré.
- Cliquez sur **OK**. Une réaction est ajoutée au noeud Relations dans l'arbre de spécifications.

- Enregistrez le fichier et fermez-le.

4. Enregistrez le fichier et fermez-le.

5. Ouvrez le fichier Container.CATProduct, sélectionnez la commande **Insertion->Composant existant...**. La boîte de dialogue Sélection de fichier s'affiche. Sélectionnez le fichier Plunger.CATProduct, puis cliquez sur **Ouvrir**. Le message indiqué à l'étape 3 s'affiche.

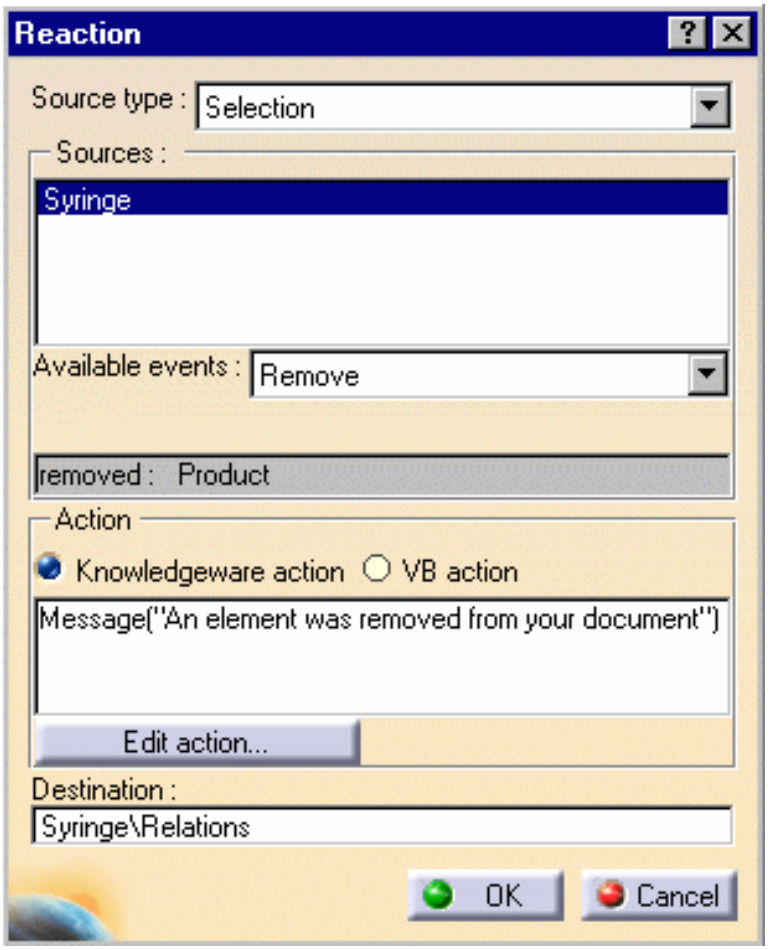


Pour plus d'informations sur la fenêtre de la fonction Réaction, reportez-vous à la section Utilisation de la fenêtre de la fonction Réaction.

Création d'une réaction : événement Remove

Dans cette tâche, vous apprendrez à utiliser l'événement Remove dans la fonction Réaction. Dans l'exemple ci-dessous, vous allez supprimer un élément dans le document CATProduct qui contient un message.

1. Ouvrez le fichier KwrSyringeAssembly2.CATProduct.
2. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier Knowledge Advisor et cliquez sur l'icône Réaction ( ) afin de créer une réaction. La boîte de dialogue Réaction s'affiche :



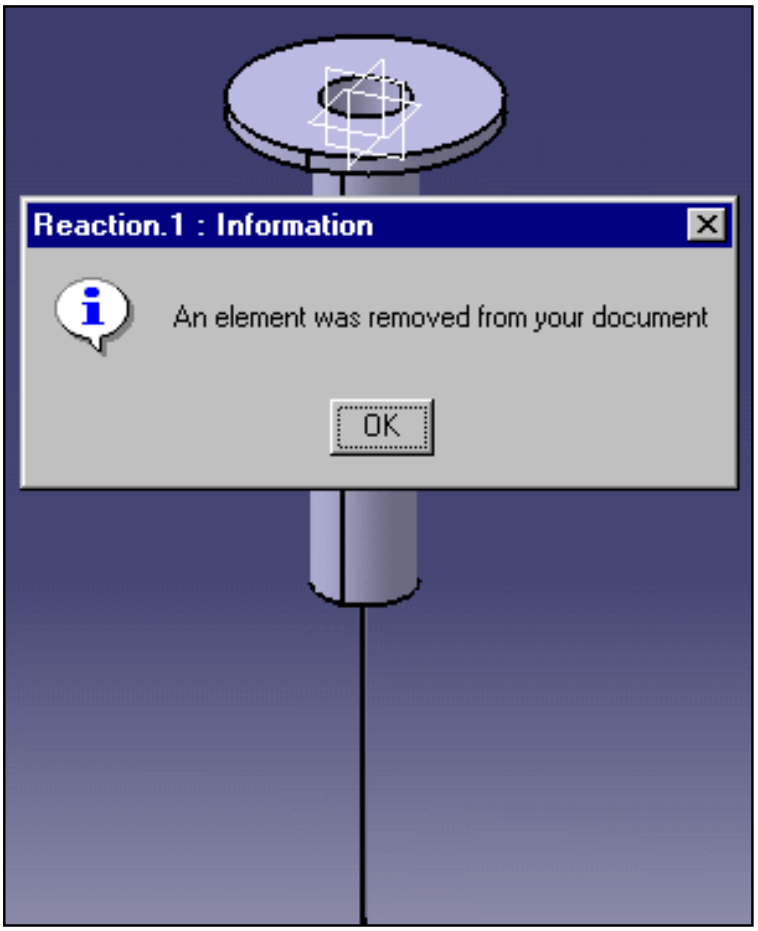
- Dans le champ **Type de la source**, sélectionnez **Sélection** pour que la réaction s'applique à l'élément sélectionné (dans cet exemple, Syringe).
- Dans la liste **Evénements disponibles**, sélectionnez **Remove** pour que la réaction se produise après la suppression

d'un élément du document CATProduct.

- Dans le champ **Action**, sélectionnez **Action Knowledgeware** et entrez le message suivant : Message("Un élément a été supprimé du document"). Ce message s'affichera chaque fois qu'un composant sera supprimé du document CATProduct.

- Cliquez sur **OK**. Une réaction est ajoutée au noeud Relations dans l'arbre de spécifications.

3. Cliquez deux fois sur la racine de l'arbre de spécifications, cliquez avec le bouton droit de la souris sur le piston Syringe dans l'arbre de spécifications puis sélectionnez **Supprimer**. Le message suivant s'affiche.



Pour plus d'informations sur la fenêtre de la fonction Réaction, reportez-vous à la section Utilisation de la fenêtre de la fonction Réaction.

Création d'une réaction : événement BeforeUpdate

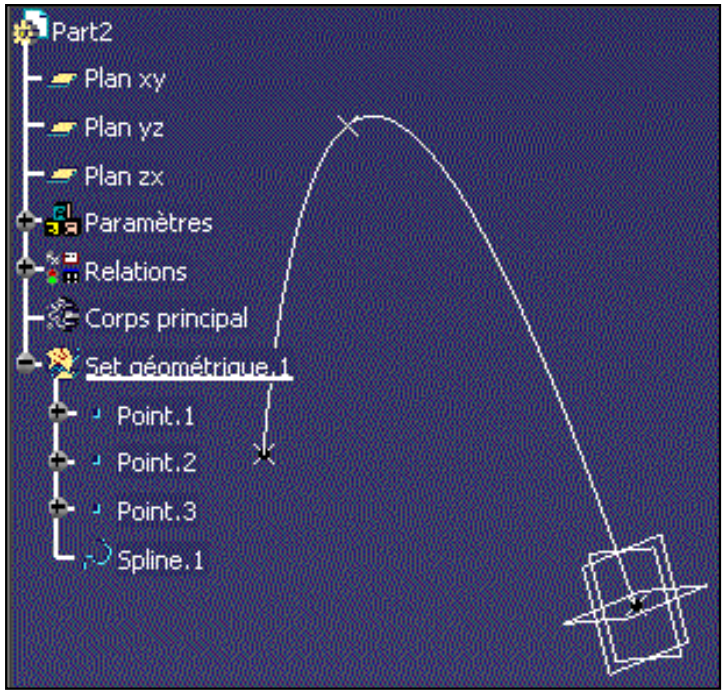
Dans cette tâche, vous apprendrez à utiliser l'événement BeforeUpdate dans la fonction Réaction. Dans l'exemple ci-dessous, vous allez optimiser la position d'un point chaque fois que vous modifiez la longueur du câble (spline). Vous créez votre géométrie et insérez tous les composants dans un gabarit (UDF). Ce gabarit contient la géométrie d'un câble passant par trois points :

- Les deux points situés aux deux extrémités doivent être définis comme entrée.
- Les coordonnées du troisième sont optimisées afin d'atteindre une longueur cible pour le câble.

Ce paramètre cible est un paramètre publié du gabarit. L'optimisation est lancée à l'aide d'une macro VB avec des arguments, appelée dans une réaction à l'événement "Before Update" du gabarit.

Ce scénario requiert Product Engineering Optimizer.

1. Ouvrez le fichier KwrEvent_BeforeUpdate.CATPart : il contient 3 points et une spline (appelée câble dans cet exemple).



2. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier **Product Engineering Optimizer** et cliquez sur l'icône **Optimisation** ( ). La fenêtre Optimisation s'affiche.
3. Entrez les données suivantes dans la fenêtre Optimisation :

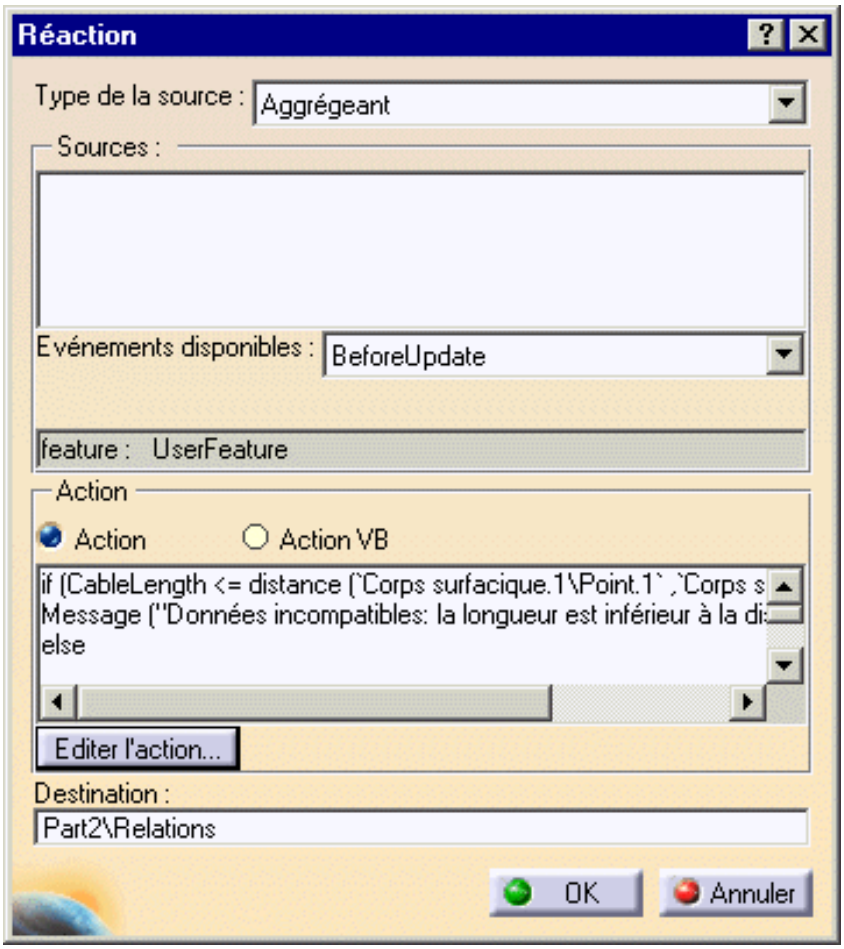
Onglet Problème		
	Type d'optimisation	**Minimisation**

	Paramètre à optimiser	**distance**
	Paramètres libres	**Geometrical Set.1\Point.3\Z**
	Algorithme	**Recuit simulé - Vitesse de convergence**
	Critère de terminaison	Nombre maximum de mises à jour : **100** Nombre de mises à jour consécutives sans amélioration : **20** Durée maximum (minutes) : **5**
Onglet Contraintes		
	Nouvelle contrainte	**Geometrical Set.1\Point.3\Z` - max (Geometrical Set.1\Point.2\Z) ,Geometrical Set.1\Point.1\Z) <= 0mm**

4. Cliquez sur **OK** dans la boîte de dialogue qui s'ouvre, puis sur **Lancer l'optimisation**.
5. Sélectionnez un fichier de sortie et cliquez sur **Enregistrer**.
6. Cliquez sur **OK** une fois le processus d'optimisation terminé.
7. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier Knowledge Advisor et cliquez sur l'icône Macros avec arguments (). L'éditeur de script s'affiche. Entrez les données suivantes dans l'éditeur et cliquez sur **OK** :

Argument	**optim**
Corps du script	**optim.Run false**

8. Cliquez sur l'icône Réaction (). La boîte de dialogue Réaction s'affiche.

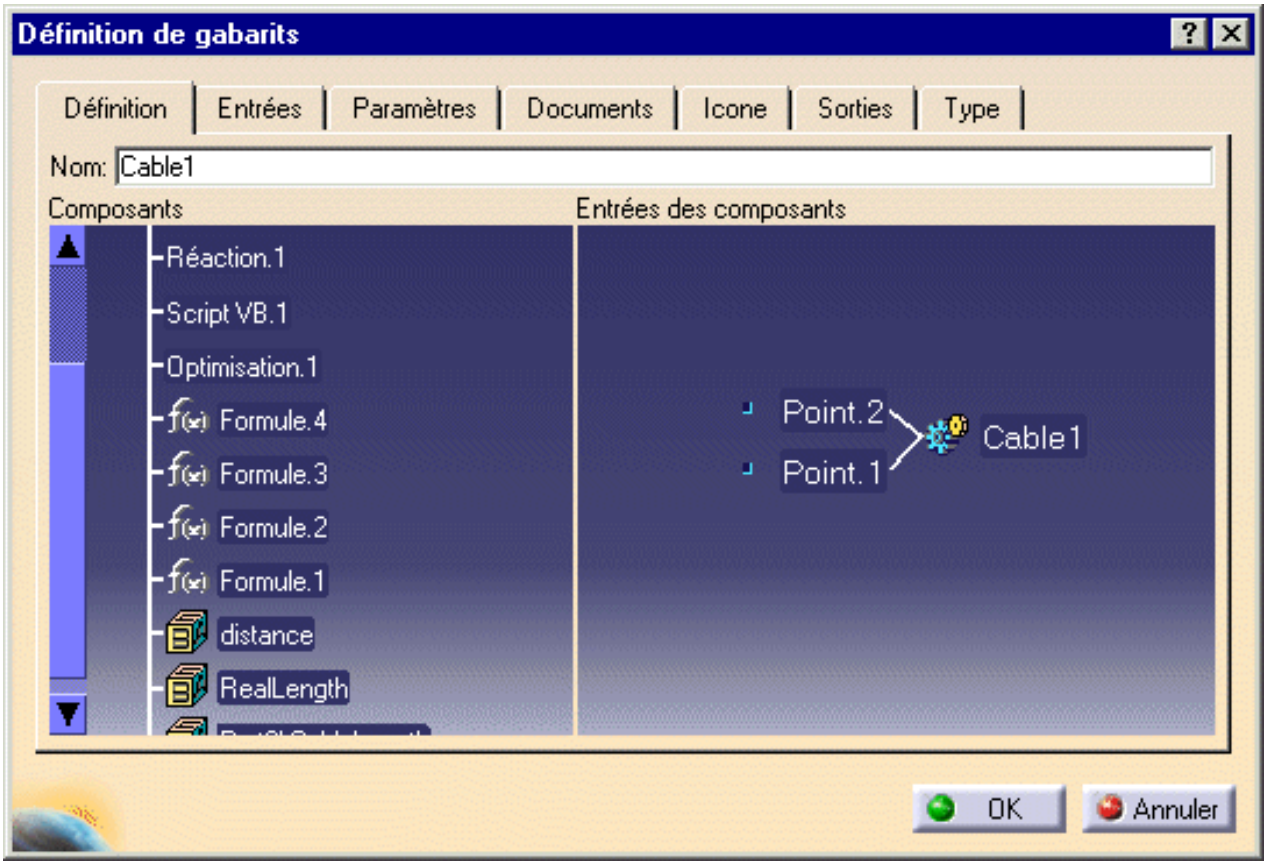


- Dans le champ **Type de la source**, sélectionnez **Agrégeant**.
- Dans la liste **Evénements disponibles**, sélectionnez **BeforeUpdate**.
- Dans le champ **Action**, sélectionnez **Action Knowledgeware**.

9. Cliquez sur le bouton **Editer l'action...**, collez le script suivant dans l'éditeur, puis cliquez deux fois sur **OK** ;

```
if (CableLength <= distance (Geometrical Set.1\Point.1 ,Geometrical Set.1\Point.2 ))
Message ("Entrées incompatibles : la longueur du câble est inférieure à la distance entre les points!")
else
{
`VB Script.1` .Run(Relations\Optimizations.1\Optimization.1 )
}
```

10. Cliquez deux fois sur la racine de l'arbre de spécifications et sélectionnez la commande **Insertion->Gabarit->Création de gabarit...**. La fenêtre Définition de gabarits s'affiche.
11. Dans le champ **Nom**, entrez le nom de l'UDF : **Cable1** dans ce scénario.
12. Sélectionnez les éléments Spline, Point3, Reaction.1, VB Script.1, Optimization.1, les quatre formules ainsi que les paramètres affichés dans la fenêtre de définition de gabarits (voir ci-dessous).



Notez que l'UDF devient le propriétaire de la réaction. Celle-ci sera déclenchée avant la mise à jour de l'instance UDF.

13. Cliquez sur l'onglet **Paramètres**, sélectionnez CableLength, cliquez sur la case **Nom** puis sur **OK**.

14. Enregistrez le fichier et fermez-le.

15. Créez un nouveau fichier .CATPart, ouvrez l'atelier Generative Shape Design et créez 3 points.

16. Sélectionnez la commande **Insertion->Instanciation...**. La fenêtre Sélection de fichier s'affiche. Sélectionnez le fichier KwrEvent_BeforeUpdate.CATPart que vous venez d'enregistrer et cliquez sur le bouton **Ouvrir**.

17. La boîte de dialogue Insertion d'objet s'affiche. Sélectionnez Point.1 et Point.2 dans la géométrie ou dans l'arbre de spécifications puis cliquez sur **OK**. Le câble (UDF) est instancié et l'optimisation est lancée avant la mise à jour.

18. Répétez les étapes 14 et 15 puis sélectionnez Point.2 et Point.3 lors de l'instanciation du gabarit : les longueurs du câble sont alors optimisées.

19. Cliquez deux fois sur le paramètre **CableLength=400mm** et remplacez sa valeur par 200

mm. La longueur de ce câble est à nouveau optimisée juste avant la mise à jour.

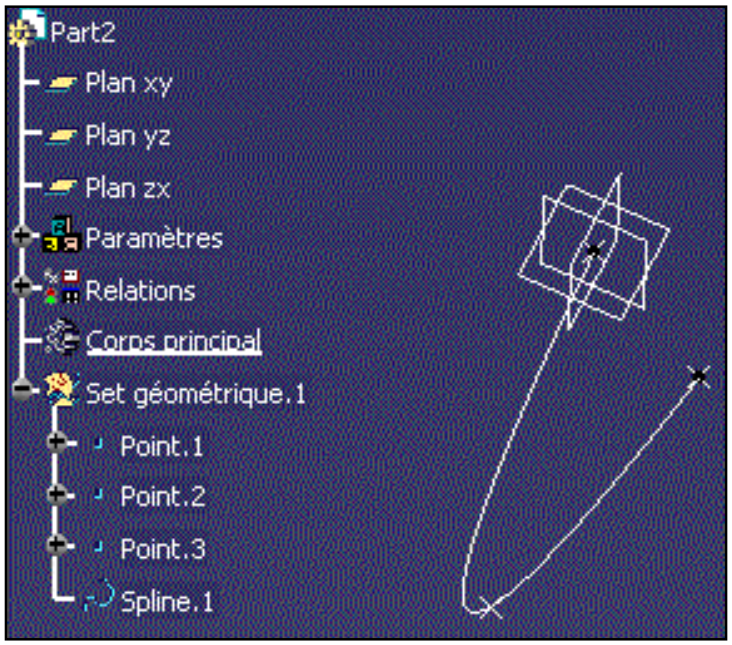
Pour plus d'informations sur la fenêtre de la fonction Réaction, reportez-vous à la section Utilisation de la fenêtre de la fonction Réaction.

Création d'une réaction : événement ValueChange

Dans cette tâche, vous apprendrez à utiliser l'événement Value associé à la fonction Réaction.
Le fichier CATPart contient un câble passant par 3 points. Vous souhaitez que la longueur du câble soit optimisée chaque fois que vous la modifiez.
L'exemple se divise en trois parties :

- Vous créez une optimisation.
- Vous créez une réaction.
- Vous modifiez la valeur de la longueur du câble.

1. Ouvrez le fichier KwrEventValueChange.CATPart : il contient 3 points et une spline (appelée câble dans ce scénario).



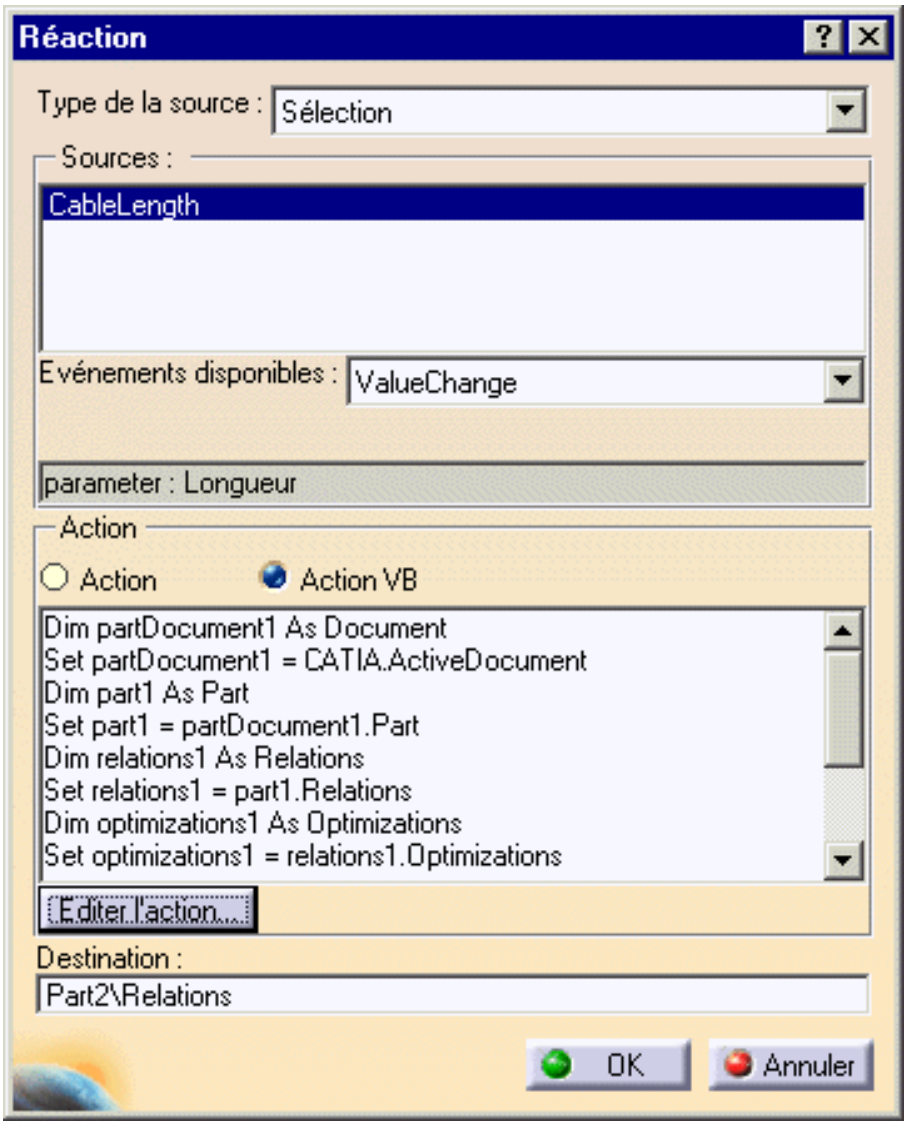
2. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier Product Engineering Optimizer et cliquez sur l'icône **Optimisation** ( ). La fenêtre Optimisation s'affiche.

3. Entrez les données suivantes dans la fenêtre Optimisation :

Onglet Problème		
	Type d'optimisation	**Minimisation**

	Paramètre à optimiser	**distance**
	Paramètres libres	**Geometrical Set.1\Point.3\Z**
	Algorithme	**Recuit simulé - Vitesse de convergence**
	Critère de terminaison	Nombre maximum de mises à jour : **100** Nombre de mises à jour consécutives sans amélioration : **20** Durée maximum (minutes) : **5**
Onglet Contraintes		
	Nouvelle contrainte	**`Geometrical Set.1\Point.3\Z` - max (Geometrical Set.1\Point.2\Z ,Geometrical Set.1\Point.1\Z) <= 0mm**

4. Cliquez sur **OK** dans la boîte de dialogue qui s'ouvre puis sur **Lancer l'optimisation**.
5. Sélectionnez un fichier de sortie et cliquez sur **Enregistrer**.
6. Cliquez sur **OK** une fois le processus d'optimisation terminé.
7. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier Knowledge Advisor et cliquez sur l'icône Réaction (). La boîte de dialogue Réaction s'affiche.



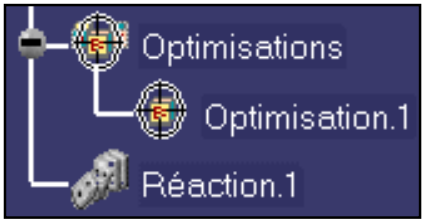
- Dans le champ **Type de la source**, sélectionnez **Sélection** ainsi que le paramètre CableLength dans l'arbre de spécifications pour que la réaction s'applique au paramètre CableLength.
- Dans la liste **Evénements disponibles**, sélectionnez **ValueChange**.
- Dans le champ **Action**, sélectionnez **Action VB**.

8. Cliquez sur le bouton **Editer l'action...**, collez le script suivant dans l'éditeur, puis cliquez deux fois sur **OK** ;

```
Dim partDocument1 As Document
Set partDocument1 = CATIA.ActiveDocument
Dim part1 As Part
Set part1 = partDocument1.Part
Dim relations1 As Relations
Set relations1 = part1.Relations
Dim optimizations1 As Optimizations
Set optimizations1 = relations1.Optimizations
Dim anyObject1 As Optimization
Set anyObject1 = optimizations1.Item("Optimization.1")

anyObject1.Run False
```

La réaction est ajoutée à l'arbre de spécifications.



9. Cliquez deux fois sur le paramètre **CableLength=400mm** et remplacez sa valeur par 600 mm : l'optimisation est lancée (le paramètre RealLength et les paramètres de distance ont été modifiés) et la géométrie est modifiée en conséquence.

Pour plus d'informations sur la fenêtre de la fonction Réaction, reportez-vous à la section Utilisation de la fenêtre de la fonction Réaction.

Utilisation d'une réaction avec un gabarit : événement Instantiation

Dans cette tâche, vous apprendrez à utiliser une réaction dans un gabarit. L'exemple se divise en deux étapes principales :

- Dans la première étape, vous créez d'abord une formule qui renvoie la longueur de la ligne. Ensuite, vous créez une réaction qui va ajouter à une liste des éléments du type longueur lors de l'instanciation du document. Pour finir, vous créez une entité définie par l'utilisateur contenant la ligne, la réaction et la formule.
- Dans la seconde étape, vous ouvrez un deuxième document et créez une règle basée sur une liste qui affichera la longueur totale. Ensuite, vous instanciez l'entité définie par l'utilisateur précédemment créée dans ce document.

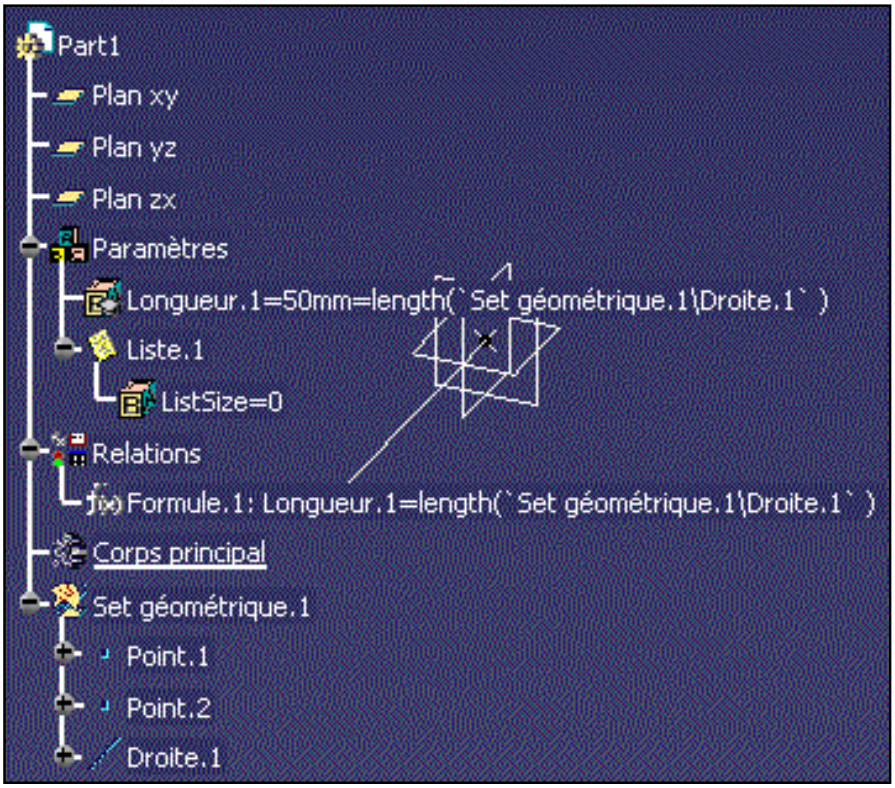
Pour réaliser cet exemple, vous devez connaître les bases du fonctionnement des ateliers Part Design et Product Knowledge Template.

Notez que l'événement Instantiation est disponible uniquement pour les gabarits. Il ne peut pas être utilisé avec les copies optimisées.

1. Ouvrez le fichier KwrUDFandReaction.CATPart.
2. Créez un paramètre de type **Longueur** et affectez-lui une formule.

   - Cliquez sur l'icône  , sélectionnez **Longueur** dans la liste déroulante **Créer paramètre de type**, cliquez sur le bouton **Créer paramètre de type**, puis cliquez sur le bouton **Ajouter formule**.
   - Dans le **Dictionnaire**, sélectionnez **Mesures**, puis cliquez deux fois sur **length (Curve,): Length**.
   - Placez le curseur entre les parenthèses et cliquez deux fois sur Line.1 dans l'arbre de spécifications. Cliquez sur **OK**, **Oui**, puis **OK**. (Length.1=length(Geometrical Set.1\Line.1)).

3. Ouvrez l'atelier Knowledge Advisor, cliquez sur l'icône **Liste** () afin de créer une liste, puis cliquez sur **OK**. Une liste vide s'affiche sous le noeud Paramètres.



4. Cliquez sur l'icône Réaction (). L'éditeur de réaction s'affiche :

   - Dans la liste **Type de la source**, sélectionnez **Agrégeant**.
   - Dans **Evénements disponibles**, sélectionnez **Instanciation**.
   - Dans la zone **Action**, sélectionnez **ActionKnowledgeware** et cliquez sur le bouton **Editer l'action**. L'éditeur d'action s'affiche.
   - Cliquez sur la liste dans l'arbre de spécifications, puis sélectionnez Liste dans la sous-fenêtre du dictionnaire et cliquez deux fois sur **List.AddItem** dans la sous-fenêtre des membres.
   - Placez le curseur entre les parenthèses, puis entrez Length.1 avant la virgule et 0 après. Cliquez deux fois sur **OK**. L'entité Réaction est créée.

5. Ouvrez l'atelier Part Design et sélectionnez la commande **Insertion->Gabarit-**

>Création d'un gabarit. La fenêtre Définition de gabarits s'affiche : dans l'onglet **Définition**, entrez le nom du gabarit (dans cet exemple, UserFeature1), puis sélectionnez les paramètres de ligne, de réaction et de longueur dans l'arbre de spécifications. Cliquez sur **OK**.

Le gabarit UserFeature1 est créé et s'affiche sous le noeud Knowledge Templates.

6. Enregistrez votre fichier, fermez-le, puis ouvrez le fichier KwrUDFandReaction2.CATPart. Il s'agit du fichier dans lequel vous allez instancier l'entité définie par l'utilisateur précédemment créée.

7. Ouvrez l'atelier Knowledge Advisor, cliquez sur l'icône de la liste () afin de créer une liste, puis cliquez sur **OK**.

8. Créez un paramètre de type **Longueur** (appelé Length.1 dans cet exemple) et affectez-lui une formule.

   - Cliquez sur l'icône  , sélectionnez **Longueur** dans la liste déroulante, cliquez sur le bouton **Créer paramètre de type**, puis sur le bouton **Ajouter formule**.

   - Cliquez sur la liste dans l'arbre de spécifications, puis sélectionnez Liste dans la sous-fenêtre du dictionnaire et cliquez deux fois sur **List.sum** dans la sous-fenêtre des membres. Cliquez trois fois sur **OK**.

9. Cliquez sur l'icône Règle (), puis sur **OK** et entrez le script suivant dans l'éditeur de règles. Cliquez ensuite sur **OK** :

```
Message("Total Length : #",Length.1 )
```

   La longueur totale s'affiche : 0 mm.

10. Ouvrez l'atelier Part Design et sélectionnez la commande **Insertion->Instanciation**.

11. Sélectionnez le fichier que vous avez créé (de l'étape 1 à l'étape 5, appelé KwrUDFandReaction.CATPart dans cet exemple), puis cliquez sur **Ouvrir**. La fenêtre **Insertion d'objet** s'affiche :

- Sélectionnez Point.1 dans l'arbre de spécifications ou dans la géométrie.

- Sélectionnez Point.2 dans l'arbre de spécifications ou dans la géométrie.

- Sélectionnez List.1 dans l'arbre de spécifications, puis cliquez sur **OK**. La règle est déclenchée et la longueur totale s'affiche.

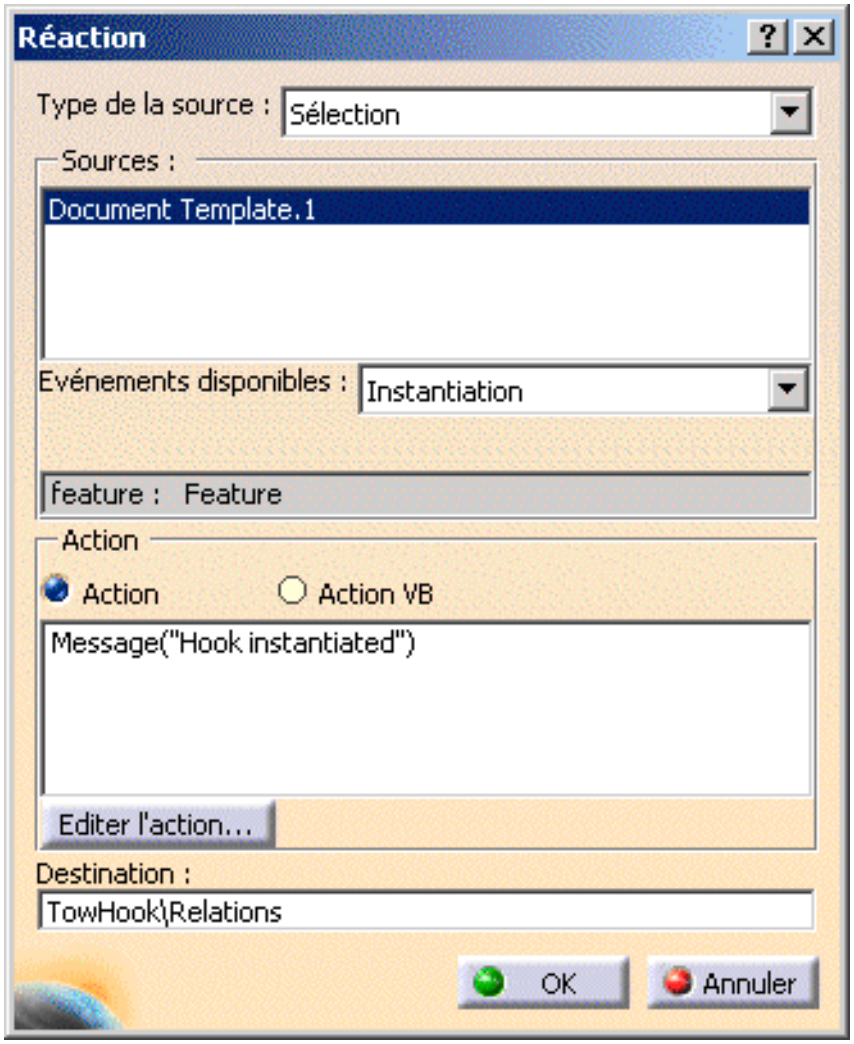


Utilisation d'une réaction avec un modèle de document : événement Instantiation

Dans cette tâche, vous apprendrez à utiliser l'événement Instanciation associé à la fonction Réaction. Dans ce scénario, vous allez instancier un modèle de document contenant un clavier dans un fichier .CATProduct contenant déjà un support de téléphone mobile.

1. Ouvrez le fichier KwrTowHook.CATProduct. Notez que le document contient déjà un modèle de document.

2. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier **Knowledge Advisor** et cliquez sur l'icône **Réaction** (). La boîte de dialogue Réaction s'affiche.



- Dans le champ **Type de la source**, sélectionnez **Sélection** et cliquez sur Document Template.1 sous le noeud Knowledge Templates.
- Dans la liste **Evénements disponibles**, sélectionnez **Instantiation**.
- Dans le champ **Action**, sélectionnez **Action Knowledgeware** et entrez le message suivant : Message("Crochet instancié"). Cliquez sur **OK** lorsque vous avez terminé. L'entité Réaction s'ajoute au noeud Relations.

3. Cliquez sur **OK** lorsque vous avez terminé. La réaction est ajoutée à l'arbre de spécifications.
4. Enregistrez le fichier, puis fermez-le.
5. Ouvrez le fichier KwrDestinationProduct.CATProduct.
6. A partir du menu **Démarrer->Knowledgeware**, ouvrez l'atelier **Product Knowledge Template**.
7. Cliquez sur l'icône **Instancier à partir d'un document...**.
8. Dans la boîte de dialogue **Sélection de fichier**, sélectionnez le fichier KwrTowHook.CATPart que vous venez d'enregistrer, puis cliquez sur **Ouvrir**. La boîte de dialogue Insertion d'objet s'affiche.
9. Sélectionnez les éléments appropriés dans la géométrie, puis cliquez sur **OK** lorsque vous avez terminé. Le modèle de document est instancié et la réaction est lancée.



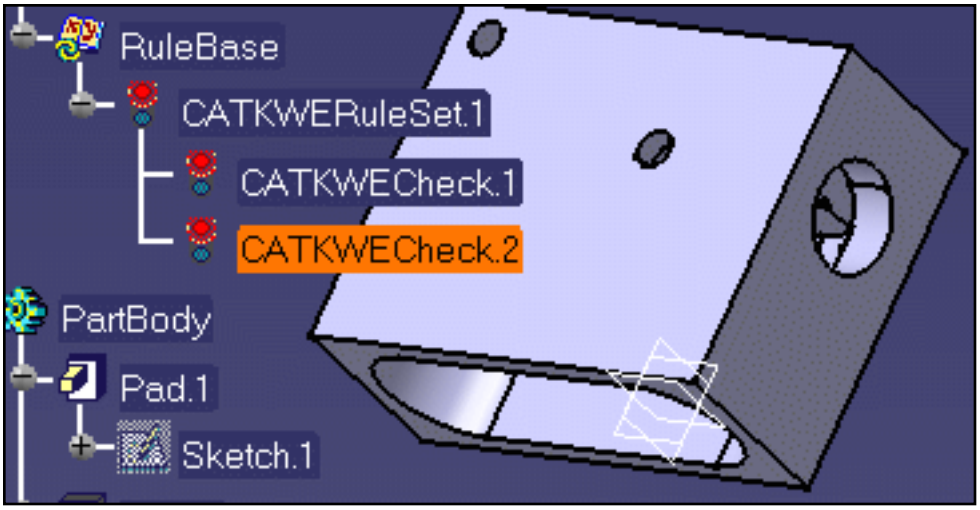
Pour plus d'informations sur la fenêtre de la fonction Réaction, reportez-vous à la section Utilisation de la fenêtre de la fonction Réaction.



Création d'une réaction : événement Update

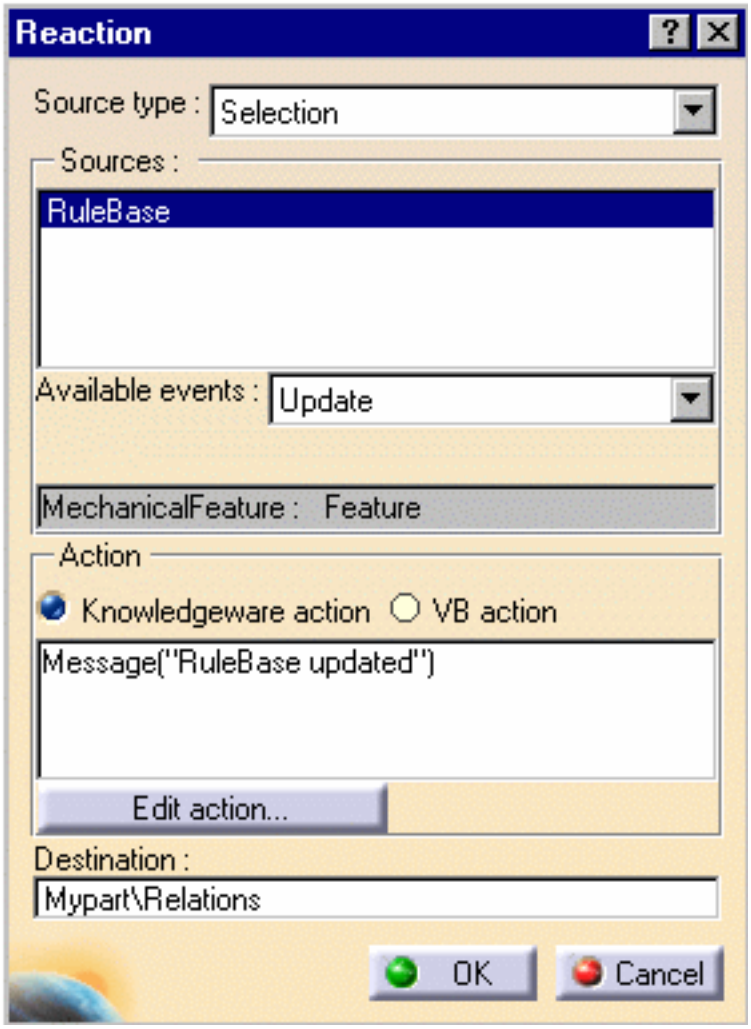
Dans cette tâche, vous apprendrez à utiliser l'événement Update associé à la fonction Réaction.
Le fichier CATPart contient une base de règles mise à jour chaque fois qu'une modification est effectuée.

1. Ouvrez le fichier KwrEvent_Update.CATPart : il contient une pièce avec des trous et une base de règles composée de deux vérifications.



2. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier Knowledge Advisor et cliquez sur l'icône Réaction (). La boîte de dialogue Réaction s'affiche.

- Dans le champ **Type de la source**, sélectionnez **Sélection** puis le paramètre RuleBase dans l'arbre de spécifications pour que la réaction s'applique à la base de règles.



- Dans la liste **Evénements disponibles**, sélectionnez **Update**.
- Dans le champ **Action**, sélectionnez **Action Knowledgeware** et entrez le message suivant : Message("Base de règles mise à jour").

3. Cliquez deux fois sur CATKWECheck.1. L'éditeur de vérifications s'affiche. Modifiez la vérification : (H\Diameter == 20mm) puis cliquez sur **OK**.
4. Cliquez avec le bouton droit de la souris sur la base de règles et sélectionnez la commande **Objet Base de règles->Résolution manuelle complète**. La réaction est déclenchée et le message suivant s'affiche :

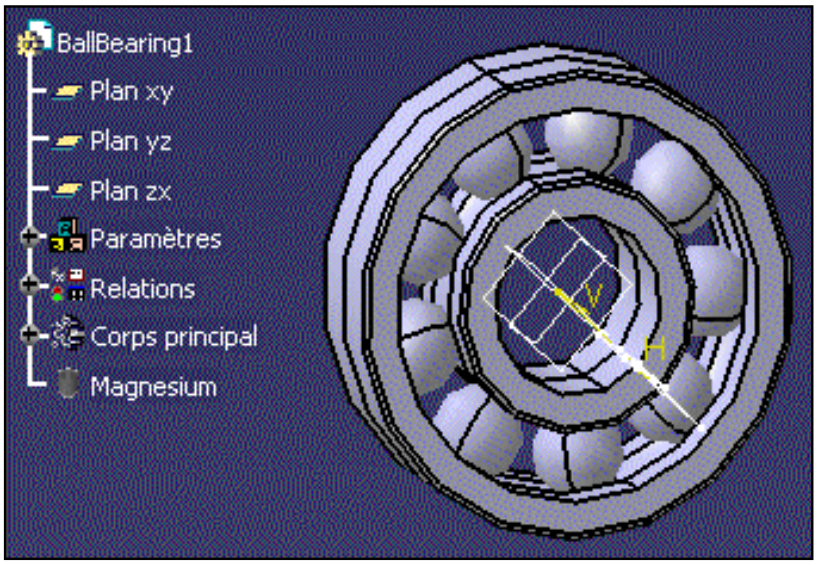


Pour plus d'informations sur la fenêtre de la fonction Réaction, reportez-vous à la section Utilisation de la fenêtre de la fonction Réaction.

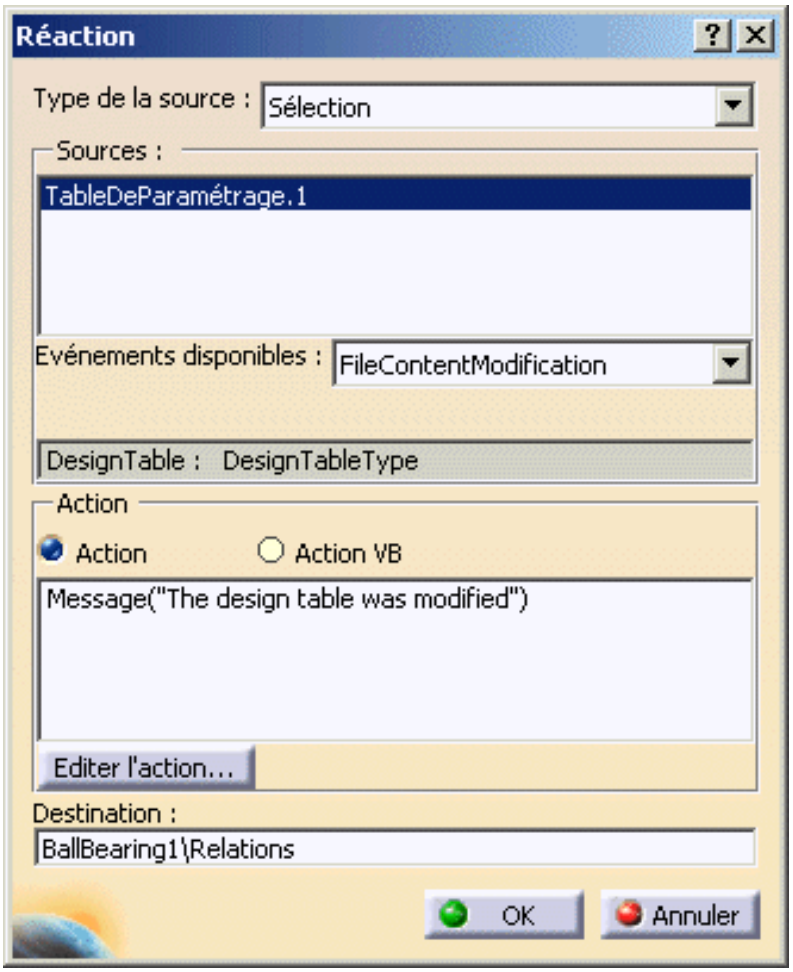
Création d'une réaction : événement FileContentModification

Dans cette tâche, vous apprendrez à utiliser l'événement FileContentModification associé à la fonction Réaction. Cet événement déclenche une réaction chaque fois que le fichier associé à la table de paramétrage est modifié.

1. Ouvrez le fichier KwrBallBearing1.CATPart. L'image suivante s'affiche.



2. Cliquez sur l'icône **Table de paramétrage** ( ). La boîte de dialogue Création d'une table de paramétrage s'affiche.
3. Sélectionnez l'option **Créer une table de paramétrage à partir d'un fichier existant** puis cliquez sur **OK**. La boîte de dialogue Sélection de fichier s'affiche.
4. Sélectionnez le fichier KwrBearingDesignTable.xls puis cliquez sur **Ouvrir**. Cliquez sur **Oui** lorsque l'on vous demande si vous souhaitez associer les colonnes des tables aux paramètres.
5. Cliquez sur **OK** pour appliquer la configuration par défaut.
6. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier Knowledge Advisor et cliquez sur l'icône **Réaction** ( ). La boîte de dialogue Réaction s'affiche.



- Dans le champ **Type de la source**, sélectionnez **Sélection** ainsi que le paramètre DesignTable.1 dans l'arbre de spécifications pour que la réaction s'applique à la table de paramétrage.
- Dans la liste **Evénements disponibles**, sélectionnez **FileContentModification**.
- Dans le champ **Action**, sélectionnez **Action Knowledgeware** et entrez le message suivant : Message("La table de paramétrage a été modifiée"). Cliquez sur **OK** lorsque vous avez terminé. L'entité Réaction s'ajoute au noeud Relations.

7. Dans l'arbre de spécifications, cliquez deux fois sur DesignTable.1. La fenêtre Table de paramétrage s'affiche.
8. Cliquez sur le bouton **Modifier la table...** et remplacez la valeur contenue sur la ligne 2 par Gold. Enregistrez le fichier, puis fermez-le. La réaction est déclenchée et le message suivant s'affiche.



Pour plus d'informations sur la fenêtre de la fonction Réaction, voir Utilisation de la fenêtre de la fonction Réaction.

Création d'une boucle dans une réaction

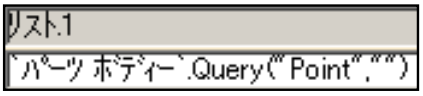
Dans cette tâche, vous apprendrez à créer une boucle dans une réaction. Dans ce scénario, vous allez :

- créer 3 points qui sont insérés dans une liste ;
- créer une formule qui sera automatiquement ajoutée à la liste ;
- créer une réaction qui fixera la coordonnée Z des points à 20 mm si cette valeur est inférieure à 20 mm ;
- créer une réaction qui fixera la coordonnée Y des points à 40 mm si cette valeur est inférieure à 40 mm ;
- mettre à jour la pièce.

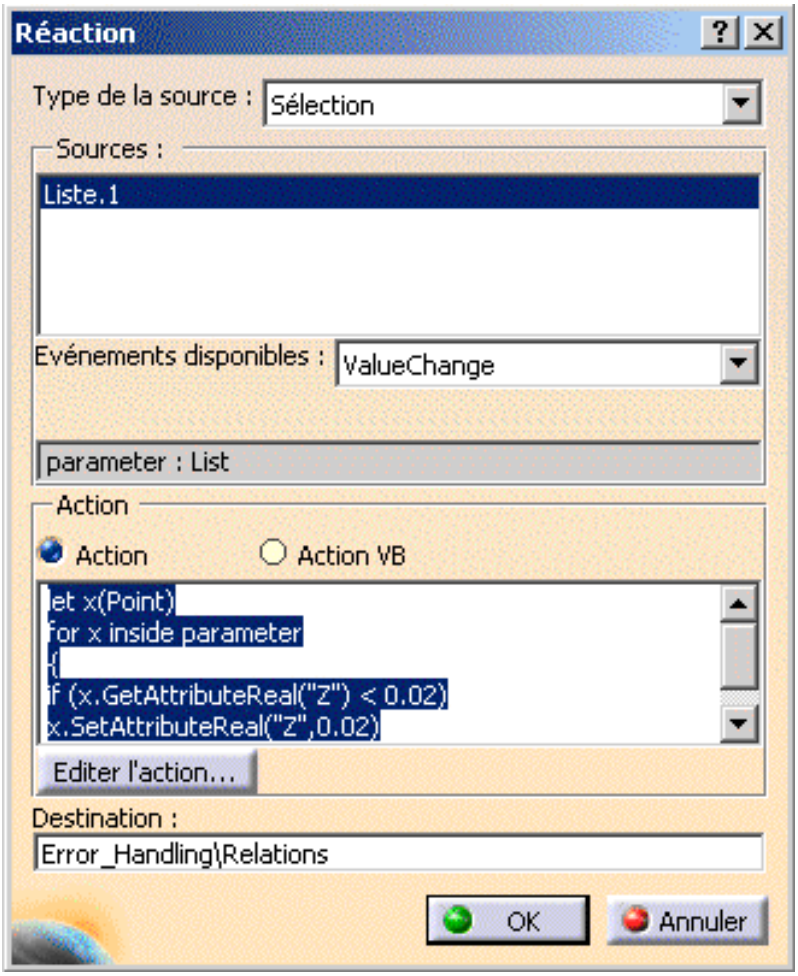
Il est désormais possible de créer des constructions While et For. Pour plus d'informations, voir Utilisation du langage Knowledge Advisor.

1. Créez une pièce contenant 3 points.
2. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier **Knowledge Advisor**.
3. Cliquez sur l'icône **Liste** () pour créer la liste qui contiendra les points créés.
4. Ajoutez une formule à la liste.
   - Cliquez sur l'icône Editeur de formules. L'éditeur de formules s'affiche. Dans l'arbre de spécifications, sélectionnez List.1 et cliquez sur le bouton **Ajouter formule**.
   - List.1=PartBody.Query("Point",""). Lorsque vous avez terminé, cliquez à deux reprises sur **OK**.

   Les utilisateurs japonais doivent entrer la formule suivante :

6. Cliquez sur l'icône **Réaction**(). La boîte de dialogue Réaction s'affiche.



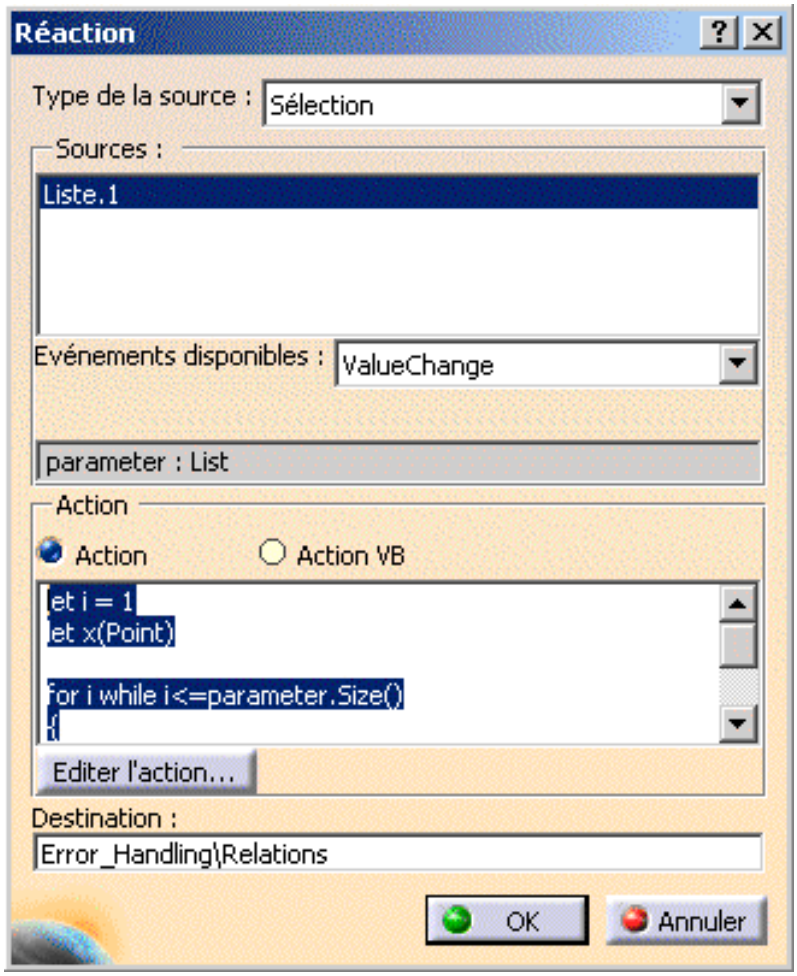
- Dans le champ **Type de la source**, sélectionnez **Sélection**, puis la liste dans l'arbre de spécifications pour que la réaction s'applique lorsque la liste est modifiée.
- Dans la boîte de dialogue Sélection de la source..., sélectionnez List.1 et cliquez sur **OK**.
- Dans la liste **Evénements disponibles**, sélectionnez **ValueChange**.
- Dans le champ **Action**, sélectionnez **Action Knowledgeware**, puis entrez le corps d'action suivant.

```
let x(Point)
for x inside parameter
{
if (x.GetAttributeReal("Z")
< 0.02)
x.SetAttributeReal("Z",0.02)
}
```

- Lorsque vous avez terminé, cliquez sur **OK**. L'entité Réaction s'ajoute au noeud Relations.

7. Cliquez sur l'icône **Réaction**(). La boîte de dialogue Réaction s'affiche.

- Dans le champ **Type de la source**, sélectionnez **Sélection**, puis la liste dans l'arbre de spécifications pour que la réaction s'applique lorsque la liste est modifiée.
- Dans la boîte de dialogue Sélection de la source..., sélectionnez List.1 et cliquez sur **OK**.
- Dans la liste **Evénements disponibles**, sélectionnez **ValueChange**.



Dans le champ **Action**, sélectionnez **Action Knowledgeware**, puis entrez le corps d'action suivant.

```
let i = 1
let x(Point)

for i while
i<=parameter.Size()
{
x = parameter.GetItem(i)
if (x.GetAttributeReal("Y")
< 0.04)
x.SetAttributeReal("Y",0.04)
}
```

- Lorsque vous avez terminé, cliquez sur **OK**. L'entité Réaction s'ajoute au noeud Relations.

8. Créez un autre point avec les coordonnées suivantes :

   X : 36 mm

   Y : 12 mm

   Z : 0 mm

9. Mettez deux fois à jour la pièce : les coordonnées Y et Z prennent les valeurs 40 et 20 mm. Cliquez ici pour afficher le modèle obtenu.

Pour plus d'informations sur la fenêtre de la fonction Réaction, reportez-vous à la section Utilisation de la fenêtre de la fonction Réaction.

Lancement d'une macro VB avec des arguments

La tâche ci-dessous illustre comment ajouter des arguments à une macro.

Les macros avec des arguments sont des entités qui peuvent être :

- stockées dans des documents CATPart ou CATProduct.
- stockées dans des catalogues. Cliquez deux fois dessus dans le catalogue pour les lancer.
- appelées à partir d'une règle (VBScriptRun) ou d'une réaction. Dans ce cas, les arguments sont transmis via la règle.

L'icône  permet d'accéder à un éditeur de macro. Outre les fonctions usuelles Editer et Exécuter, cet éditeur permet de :

- définir des arguments ;
- transposer la définition d'une entité dans l'éditeur. Pour cela, il suffit de sélectionner l'entité dans l'arborescence ou dans la zone géométrique.

Notez que la macro VB avec des arguments est destinée à vous faciliter la création de macros. Elle n'a pas pour objectif de rechercher les codes des autres objets : seuls les objets qui ont été correctement intégrés aux macros sont pris en charge.

1. Cliquez sur l'icône  . L'éditeur de script apparaît.
2. Copiez le script ci-dessous et collez-le dans l'éditeur :

```
Dim oActiveDoc As Document
Set oActiveDoc = CATIA.ActiveDocument

If (InStr(oActiveDoc.Name,".CATPart")) <> 0 Then

Dim oParams As Parameters
Set oParams = oActiveDoc.Part.Parameters

Dim strParam1 As StrParam
Set strParam1 = oParams.CreateString("prénom",
"")

Dim strParam2 As StrParam
Set strParam2 = oParams.CreateString("nom", "")

strParam1.Value = fname
strParam2.value = lname
```

```
Else MsgBox "Le document actif doit être de type CATPart"
End If

End Sub
```

3. Entrez les arguments fname et lname dans le champ situé entre les parenthèses. Les arguments doivent toujours être séparés par une virgule.
4. Cliquez sur OK pour ajouter la macro au document. Un sous-noeud appelé VB Scripts est ajouté dans l'arbre de spécifications sous le noeud Relations. Un objet VB Script est ajouté en dessous de ce noeud.
5. Cliquez deux fois sur l'objet VB Script. L'éditeur de script apparaît. Le bouton Insérer la résolution d'un objet permet d'extraire une définition d'entité. La macro VB Script.2 de l'exemple KwrObject.CATPart illustre comment utiliser cette fonction.
6. Cliquez sur **Exécuter script...**. La fenêtre Sélection des arguments du script s'affiche.
7. Le cas échéant, sélectionnez fname dans la liste des arguments et entrez une chaîne dans la zone de valeur (sans guillemets). Sélectionnez ensuite l'argument lname et entrez sa valeur.
8. Cliquez sur OK pour exécuter le script.Les deux paramètres de type chaîne sont ajoutés au document. Leurs valeurs sont celles que vous venez de définir.

Remarquez que les fonctions du script VB avec arguments sont fournies avec un menu contextuel permettant à l'utilisateur d'exécuter le script.

Utilisation des relations

Les relations situées au niveau d'un produit ne sont pas intégrées au processus de mise à jour du produit. Si votre relation porte des paramètres situés au niveau d'une pièce, créez la relation au niveau de la pièce.



Cliquez sur l'icône Ajout d'ensemble de relations pour regrouper les relations en catégories.



Cliquez sur l'icône Mise à jour des mesures pour effectuer la mise à jour des relations en utilisant des mesures.

Création d'ensembles de relations
Utilisation des relations basées sur les publications au niveau du produit
Activation et désactivation d'un composant
Instanciation de relations issues d'un catalogue
Mise à jour de relations utilisant des mesures
Contrôle de la mise à jour des relations

Création d'ensembles de relations

Dans cette tâche, vous apprendrez à créer des ensembles de relations.

Cette fonction permet de regrouper des relations en catégories. Lorsque vous créez une relation, vous êtes invité à entrer une *cible*. Par exemple, vous pouvez ajouter une nouvelle relation à une entité.

Les formules, les tables de paramétrage, les règles et les vérifications peuvent toutes être créées dans des ensembles de relations. Lorsqu'aucun ensemble de relations a été créé, le champ cible de l'éditeur de relations est par défaut le noeud Relations.

1. Créez une pièce puis, à partir du menu **Démarrer->Knowledgeware**, ouvrez l'atelier **Knowledge Advisor**.
2. Cliquez sur l'icône  puis sur le noeud Relations. L'ensemble de relations Relations.1 (ou Relations.*n*) est ajouté à l'arbre de spécifications juste en dessous du noeud Relations.
3. Cliquez sur l'icône Création d'une règle ou Création d'une vérification. La première boîte de dialogue qui s'affiche ressemble à celle qui apparaît lors de la création d'une relation juste en dessous du noeud Relations, si ce n'est que vous devez indiquer une cible.
   Pour ce faire, sélectionnez la valeur indiquée dans le champ **Destination** de l'éditeur de relations, puis l'ensemble de relations auquel vous souhaitez ajouter une relation. Le chemin cible dans l'éditeur de relations est donc modifié (*nom_pièce*\ Relations.*n* est remplacé par *nom_pièce*\Relations \ Relations.*n*).
4. Cliquez sur **OK** pour afficher la boîte de dialogue suivante et entrer le corps de la relation.
5. Une fois la nouvelle relation définie, cliquez sur **OK** dans la boîte de dialogue de l'éditeur. Dans l'arbre de spécifications, vous pouvez développer l'entité représentant l'ensemble de relations. Une nouvelle relation a été ajoutée

sous cet ensemble de relations.



Utilisation de relations basées sur des publications au niveau des produits

Dans cette tâche, vous apprendrez à utiliser les relations basées sur des publications au niveau des produits. Le scénario ci-après se compose des étapes suivantes :

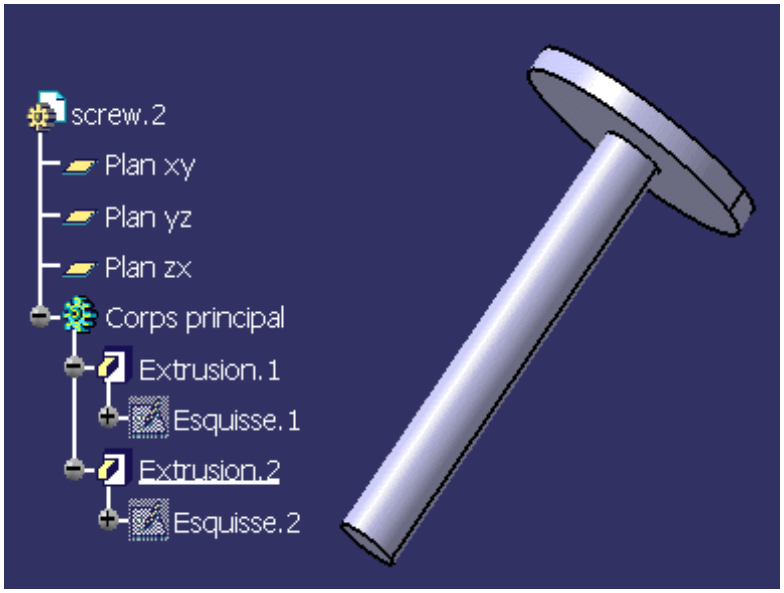
- Ajout d'un paramètre au documentKwrScrew.CATPart nommé Screw_Volume, ajout d'une formule permettant de calculer le volume de la vis et publication du paramètre Screw_Volume.
- Ajout d'un paramètre au document KwrScrew1.CATPart nommé Screw_Volume, ajout d'une formule permettant de calculer le volume de la vis et publication du paramètre Screw_Volume.
- Création d'un fichier CATProduct nommé Bolt et importation du documentKwrScrew.CATPart
- Importation du document KwrNut.CATPart.
- Dans le contexte du boulon, créez une formule permettant de calculer son volume en fonction des publications de la vis et de l'écrou.
- Dans le contexte du boulon, remplacezKwrScrew.CATPart par KwrScrew1.CATPart. Le volume est recalculé.

Avant d'exécuter ce scénario, assurez-vous que l'option **Garder le lien avec l'objet sélectionné** est activée (**Outils -> Options -> Part Infrastructure ->Général**).

Cette fonction peut être utilisée avec :

- Tables de paramétrage
- Formules
- Règles et recommandations
- Ensemble d'équations
- Optimisation

1. Ouvrez le document KwrNewScrew.CATPart. L'image suivante s'affiche.



2. Ajoutez un paramètre de volume à la pièce. Pour ce faire, procédez comme suit :

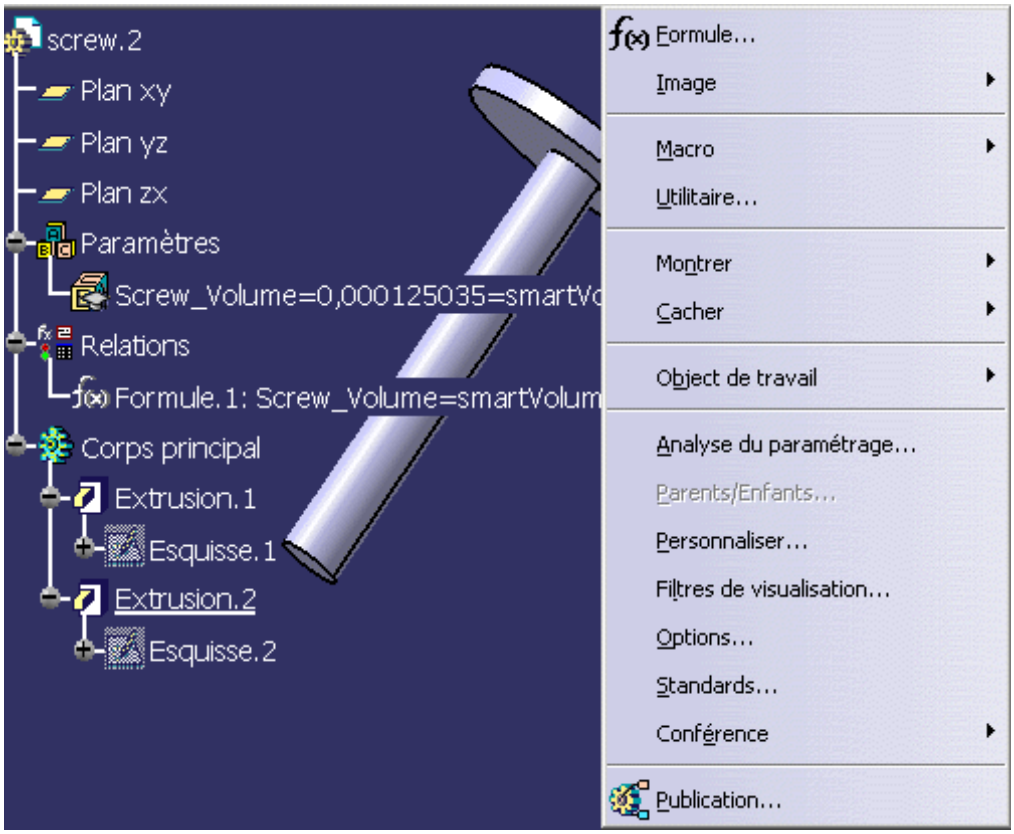
- Cliquez sur l'icône . L'éditeur de formules s'ouvre.Dans la liste déroulante **Créer paramètre de type**, sélectionnez **Réel**, puis cliquez sur le bouton **Créer paramètre de type**.

- Dans le champ **Editer le nom ou la valeur du paramètre sélectionné**, entrez le nom du paramètre : **Screw_Volume**. Cliquez sur **Appliquer**, puis sur le bouton **Ajouter formule**. L'éditeur de formules s'ouvre.

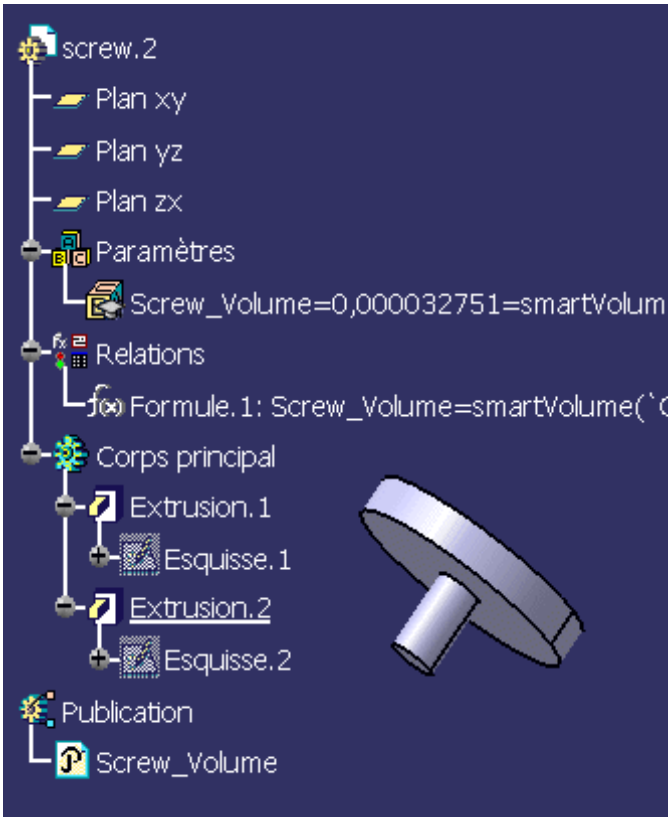
- Entrez la formule suivante à l'aide du dictionnaire : **Screw_Volume=smartVolume(PartBody\Pad.1)+smartVolume(PartBody\Pad.2)**. Cliquez trois fois sur **OK**.

3. Publiez le paramètre **Screw_Volume**.

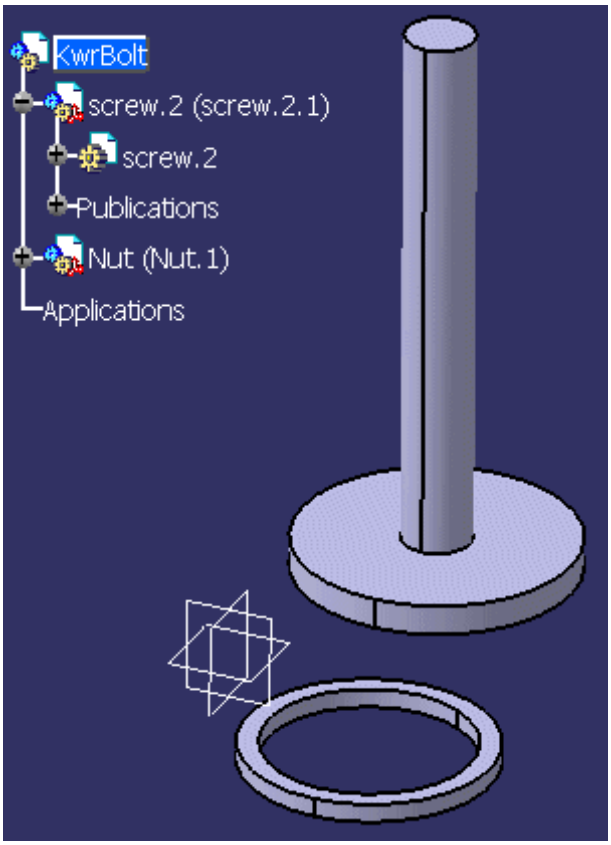
Pour ce faire, sélectionnez la commande**Outils -> Publication** et cliquez sur le paramètre **Screw_Volume** sous le noeud Paramètres dans l'arbre de spécifications. Cliquez sur **OK**. Le paramètre publié apparaît dans l'arbre de spécifications sous le noeud Publication. Enregistrez le fichier, puis fermez-le.



4. Ouvrez le document KwrNewScrew1.CATPart et répétez les étapes ci-dessus (1 à 3 incluses). La pièce doit être identique à celle ci-dessous. Enregistrez le fichier, puis fermez-le.



5. Créez un fichier CATProduct nommé KwrBolt.CATProduct.
6. Cliquez sur le produit racine et sélectionnez la commande **Insertion -> Composant existant...**. La boîte de dialogue **Sélection de fichier** s'affiche. Sélectionnez le fichier KwrNewScrew.CATPart et cliquez sur **Ouvrir**. La vis est importée.
7. Choisissez la commande **Insertion -> Composant existant...**, sélectionnez le fichier KwrNewnut.CATPart et cliquez sur **Ouvrir**. L'écrou est inséré.

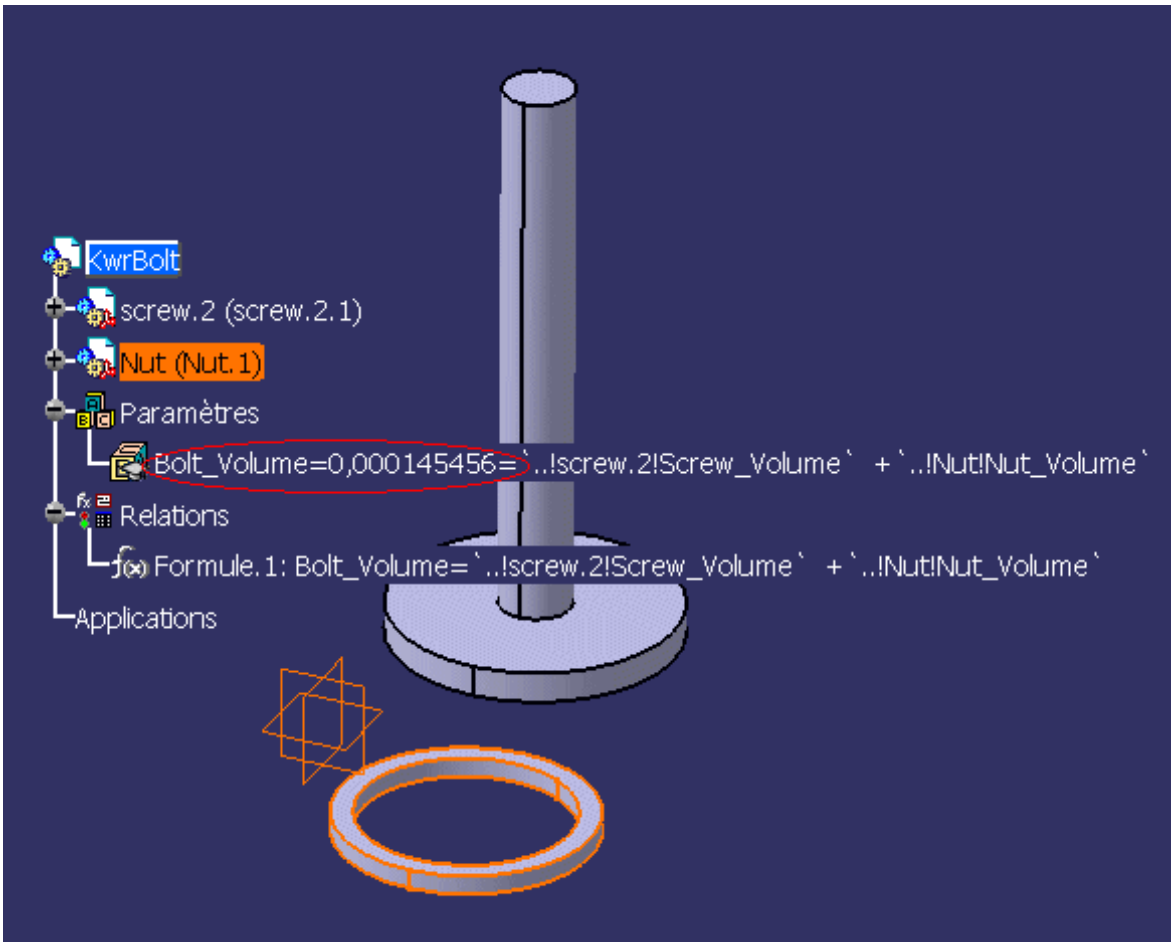


8. Ajoutez un paramètre Bolt_Volume au produit pour calculer le volume du boulon. Pour ce faire, procédez comme suit :

   - Cliquez sur le produit racine et cliquez sur l'icône  . L'éditeur de formules s'ouvre.Dans la liste déroulante **Créer paramètre de type**, sélectionnez **Réel**, puis cliquez sur le bouton **Créer paramètre de type**.

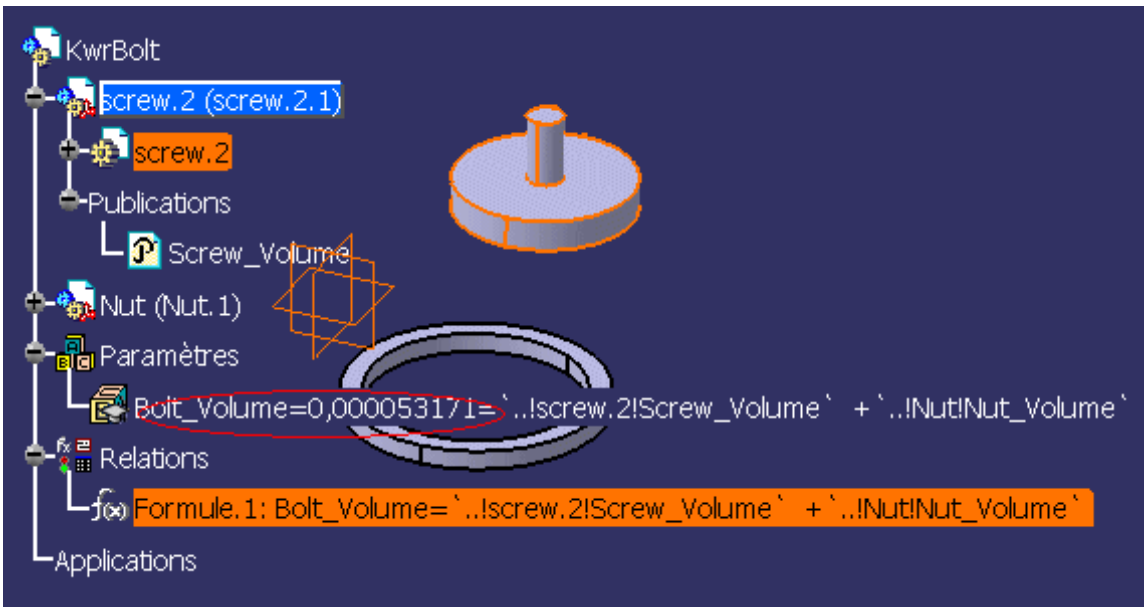
   - Dans le champ **Editer le nom ou la valeur du paramètre sélectionné**, entrez le nom du paramètre : **Bolt_Volume**. Cliquez sur **Appliquer**, puis sur le bouton **Ajouter formule**. L'éditeur de formules s'ouvre.

   - Entrez la formule suivante à l'aide du dictionnaire et en cliquant sur les paramètres publiés dans l'arbre de spécifications : **Bolt_Volume=`..!screw.2!Screw_Volume` +`..!Nut!Nut_Volume`**. Cliquez sur **OK**, et encore sur **OK**. Le volume du boulon apparaît.



9. Remplacez la vis afin de calculer un nouveau volume : cliquez deux fois, puis cliquez avec le bouton droit de la souris sur le composant Screw.2 dans l'arbre de spécifications et choisissez l'option **Composants -> Remplacer le composant...**. La boîte de dialogue **Sélection de fichier** s'affiche. Sélectionnez le fichier KwrNewScrew1.CATPart et cliquez sur **Ouvrir**.

10. Cliquez sur **Oui** et sur **OK** dans la fenêtre Les impacts au remplacement de ce produit. La nouvelle vis est insérée et le volume du boulon est mis à jour.



Activation et désactivation d'un composant

Dans cette tâche, vous apprendrez à activer et à désactiver un composant.

Dans le scénario décrit ci-après, le fichier CATProduct contient deux fichiers CATPart que vous activerez et désactiverez alternativement après la création de paramètres utilisateur et d'une règle basée sur ces paramètres.

Remarquez les situations suivantes :

- Les relations situées au niveau d'un produit ne sont pas intégrées au processus de mise à jour du produit. Si votre relation pilote les paramètres d'une pièce, créez la relation au niveau de la pièce.
- Les paramètres basés sur des règles sont conçus pour permettre à l'utilisateur de contrôler les activités des composants au niveau de l'assemblage.

Lorsque vous travaillez dans un environnement japonais, n'oubliez pas d'activer l'option **A encadrer avec le symbole '** (onglet **Outils -> Options -> Général -> Paramètres et mesures -> Knowledge**).

1. Ouvrez le fichier KwrSyringe.CATProduct et enregistrez les fichiers suivants dans le même répertoire (SyringePiston.CATPart, HollowSyringePiston.CATPart, et SyringeContainer.CATPart). Il contient une seringue constituée de trois parties distinctes : un cylindre et deux pistons différents.
2. Créez un paramètre à plusieurs valeurs de type chaîne. Pour ce faire, procédez comme suit :

- Cliquez sur l'icône  . L'éditeur de formules s'ouvre.
- Sélectionnez Chaîne dans la liste déroulante, ainsi que Valeurs multiples. Cliquez sur le bouton **Créer paramètre de type**. La boîte de dialogue **Liste de valeurs** s'affiche.
- Entrez deux valeurs différentes, Creux et Plein, puis cliquez sur **OK**.
- Modifiez le nom du nouveau paramètre (TypeSeringue dans ce scénario) dans **Editer le nom ou la valeur du paramètre sélectionné**, puis cliquez sur **OK**. Le nouveau paramètre apparaît sous le noeud **Paramètres** de l'arbre de spécifications.

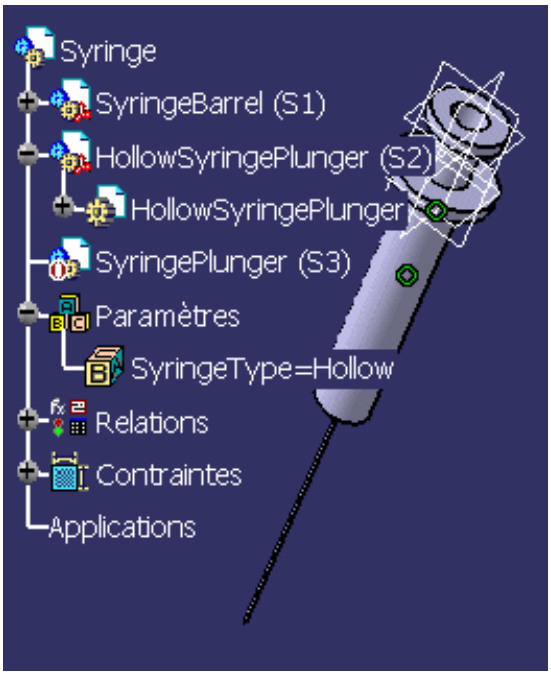
3. Accédez à l'atelier Knowledge Advisor et cliquez sur l'icône Règle pour créer une règle. Le script de cette règle vous permettra d'activer ou de désactiver l'un des pistons.
4. Entrez le code suivant dans l'éditeur de règles, puis cliquez sur **OK**.

```
if (TypeSeringue == "Creux")
{
`S3\Component Activation State` = false
`S2\Component Activation State` = true
}
else
{
`S2\Component Activation State` = false
`S3\Component Activation State` = true
}
```

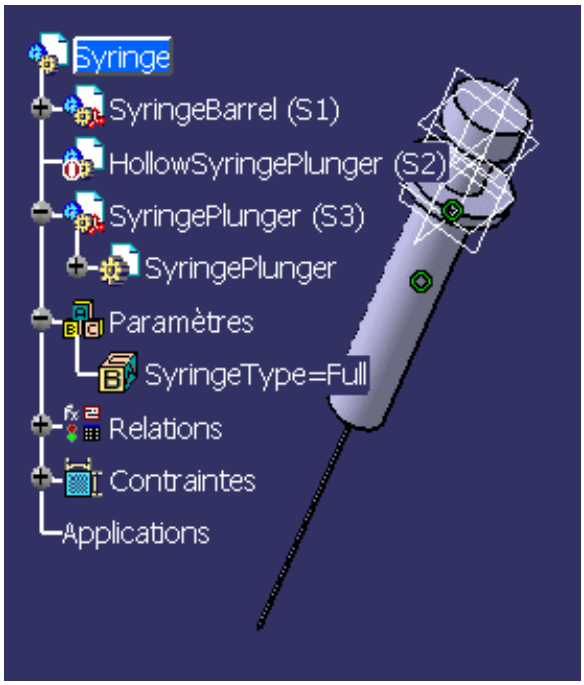
Les utilisateurs japonais doivent entrer la règle suivante :

```
if (SyringeType=="Hollow")
{
`S3¥構成要素の活動化状況` = false
`S2¥構成要素の活動化状況` = true
}
else
{
`S3¥構成要素の活動化状況` = true
`S2¥構成要素の活動化状況` = false
}
```

6. Cliquez deux fois sur le paramètre TypeSeringue sous le noeud Paramètres, puis sélectionnez Creux dans la fenêtre Editer paramètre. Les pièces SyringeBarrel CATPart et HollowSyringePlunger CATPart s'affichent.



7. Cliquez deux fois sur le paramètre TypeSeringue et sélectionnez "Plein" dans la fenêtre Editer paramètre. Les pièces SyringeBarrel CATPart et SyringePlunger CATPart s'affichent.

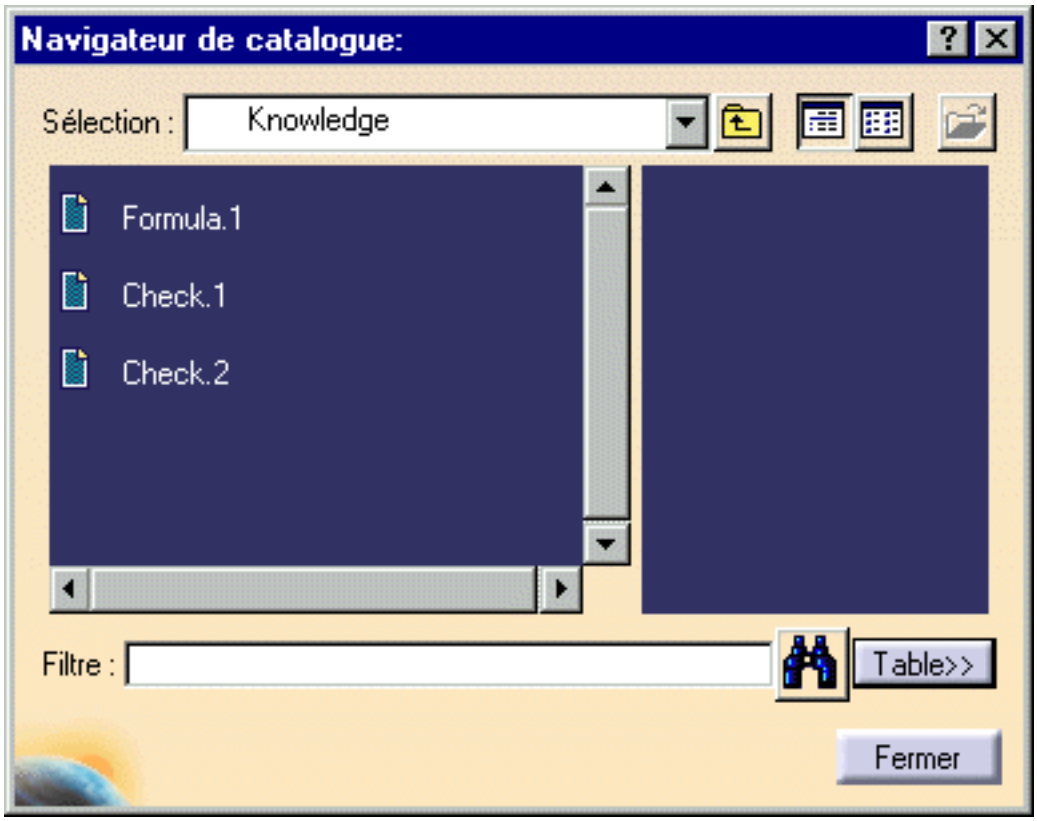


Instanciation de relations issues d'un catalogue

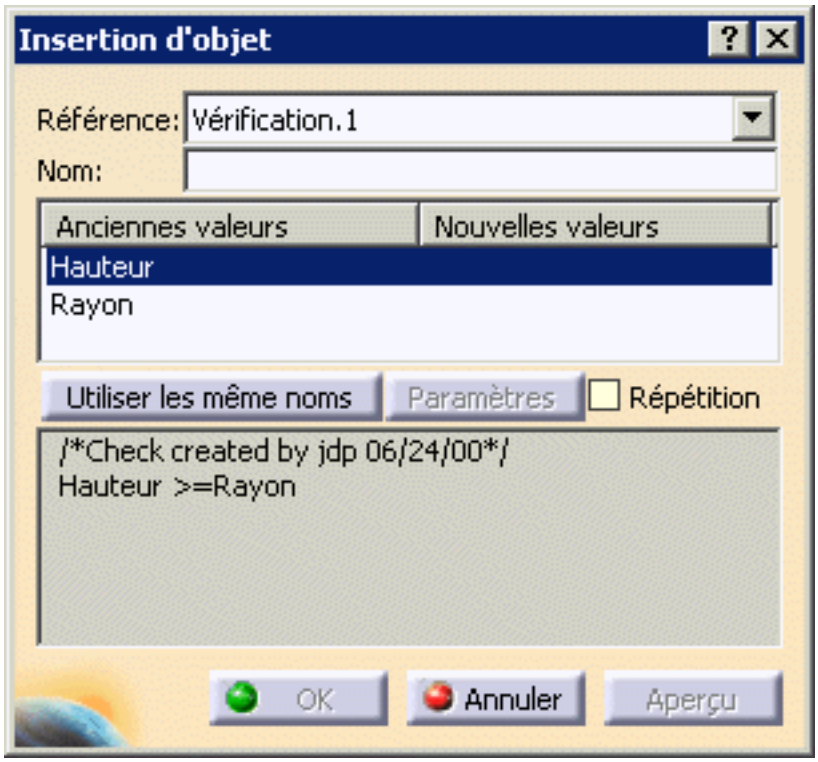
L'exemple ci-après explique comment instancier sous la forme d'un document CATPart une vérification stockée dans un catalogue.

Les formules, les règles et les vérifications peuvent être stockées dans un catalogue. Elles peuvent alors être réutilisées dans un document à l'aide d'un mécanisme d'instanciation. Pour plus d'informations sur les catalogues, reportez-vous au document *Infrastructure - Guide de l'utilisateur*.

1. Ouvrez le fichier Formula_005_Start.CATPart.
2. Dans la barre de menus Outils, cliquez sur l'icône  . Le catalogue de composant s'affiche.
3. Cliquez sur l'icône  pour ouvrir le catalogue CatalogFormula.catalog. Le catalogue de composants ressemble au catalogue présenté ci-dessous (vous pouvez développer le noeud Connaissance pour afficher les trois relations Formule.1, Vérification.1, Vérification.2. Le contenu de la partie gauche du catalogue de composants affiché dépend des dernières interactions exécutées avec cette boîte de dialogue).



4. Cliquez deux fois sur l'objet Vérification.1. La boîte de dialogue suivante s'affiche.



5. Renommez la vérification Vérification.1 dans la zone Nom. Par exemple, entrez HeightCheck.

6. L'entrée Hauteur est mise en évidence. Dans l'arbre de spécifications des

pièces, sélectionnez le paramètre "Hauteur". L'entrée Rayon est mise en évidence. Dans l'arbre de spécifications, sélectionnez le paramètre "Rayon". Cliquez sur **OK** puis sur **Fermer**. L'élément HeightCheck s'ajoute à l'arbre de spécifications et, selon les valeurs affectées aux paramètres Hauteur et Rayon, un message peut apparaître.

7. Cliquez deux fois sur la relation HeightCheck dans l'arbre de spécifications. La relation ci-après s'affiche dans l'éditeur de vérifications :

   Hauteur >= Rayon.

Les relations d'un catalogue doivent être instanciées indépendamment les unes des autres dans un document.

Mise à jour de relations utilisant des mesures

Dans cette tâche, vous apprendrez à mettre à jour des relations utilisant des mesures.

Une relation utilisant des mesures doit être mise à jour lorsque le symbole  s'affiche en face de cette relation dans l'arbre de spécifications.

Exemple

MeasureRule requiert une mise à jour.



MeasureRule ne doit pas être mise à jour.



Pour mettre à jour les règles :

- Cliquez sur l'icône  (pour cela, vous devez être dans l'atelier Knowledge Advisor)

  -ou-

- Sélectionnez la commande **Mise à jour des mesures** dans le menu contextuel du noeud Relations. Vous pouvez procéder ainsi dans tous les ateliers.

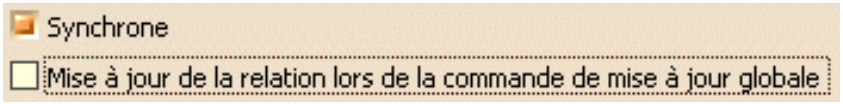
Toutes les relations du document sont alors mises à jour.

Contrôle de la mise à jour des relations

Dans cette tâche sont présentés 2 exemples courts d'utilisation de relations en modes **Synchrone** et **Mise à jour de la relation lors de la commande de mise à jour globale**.

- Dans le premier exemple, l'utilisateur met à jour une formule.
- Dans le deuxième exemple, l'utilisateur met à jour une équation.

- Pour une relation donnée, il est possible de déterminer qu'elle ne fonctionne qu'en mode synchrone à l'aide de la commande **Edition -> Propriétés**. Son évaluation peut ensuite être lancée manuellement. Le mode **synchrone** permet à l'utilisateur de créer des relations synchrones, autrement dit des relations qui seront immédiatement mises à jour si l'un de leurs paramètres est modifié. Les relations basées sur des paramètres sont les seules susceptibles d'être synchrones.



- Pour une relation donnée, il est également possible de décider que son évaluation sera lancée à la mise à jour de la pièce (**Mise à jour de la relation lors de la commande de mise à jour globale**). Les relations peuvent être asynchrones pour 2 raisons :
  - L'utilisateur veut que la relation soit asynchrone.
  - La relation contient de la géométrie.

Il est possible de changer le mode de mise à jour d'une relation après sa création. Pour ce faire, procédez comme suit :

- Cliquez avec le bouton droit de la souris dans l'arbre de spécifications et sélectionnez la commande **Propriétés**.
- Activez l'option **Synchrone** ou **Mise à jour de la relation lors de la commande de mise à jour globale** .

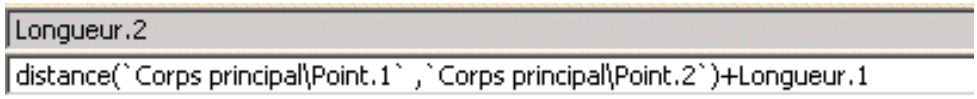
Pour plus d'informations sur les options **Synchrone** ou **Mise à jour de la relation lors de la commande de mise à jour globale**, reportez-vous à la description de l'onglet Knowledge dans le document Infrastructure - Guide de l'utilisateur.

Mise à jour d'une formule

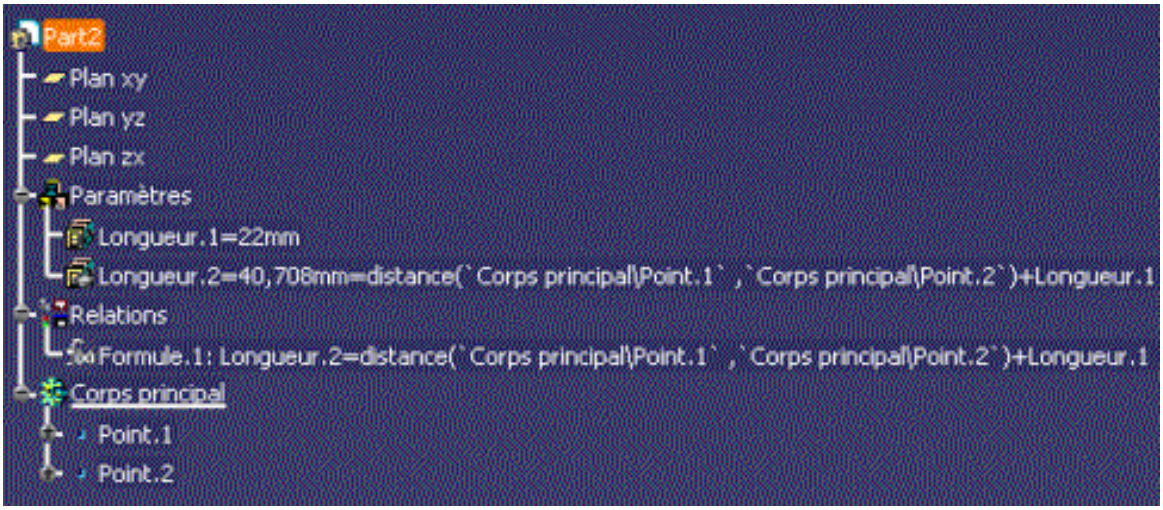
Dans cet exemple, le document contient 2 paramètres du type **longueur** :

- Longueur.1 est évalué par l'utilisateur

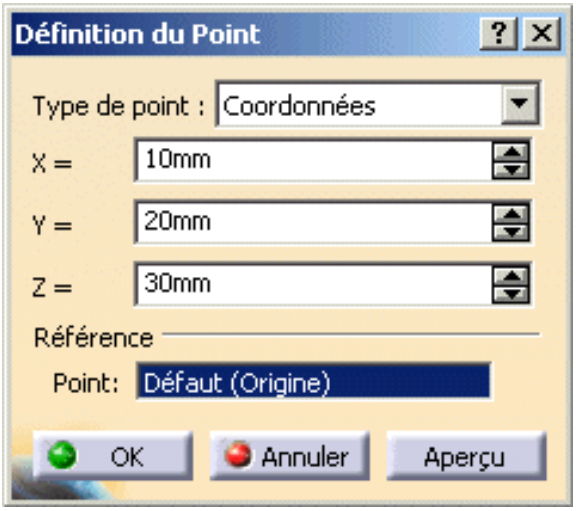
- Longueur.2 est évalué par une formule qui calcule la distance entre 2 points + Longueur.1.



1. Ouvrez le fichier KwrUpdatingaFormula.CATPart. L'image suivante s'affiche :



2. Développez le noeud Relations (si besoin est), cliquez avec le bouton droit de la souris sur Formula.1 et sélectionnez la commande**Propriétés**.
3. Dans la fenêtre **Propriétés**, désactivez l'option **Mise à jour de la relation lors de la commande de mise à jour globale**. Cliquez sur **OK** pour valider.
4. Cliquez deux fois sur Longueur.1. La fenêtre Editer paramètre s'affiche. Entrez 50 dans le champ de valeur. Cliquez sur **OK** pour valider.La valeur de Longueur.2 est automatiquement mise à jour et une icône de mise à jour s'affiche, indiquant que cette valeur sera également mise à jour lors de la mise à jour globale.
5. Cliquez avec le bouton droit de la souris sur Formula.1 et sélectionnez la commande **Propriétés**.
6. Dans la fenêtre Propriétés, désactivez l'option **Synchrone** et activez l'option **Mise à jour de la relation lors de la commande de mise à jour globale**. Cliquez sur **OK** pour valider.
7. Cliquez deux fois sur Point.2 dans l'arbre de spécifications ou dans la géométrie.
8. Entrez les valeurs suivantes dans la fenêtre **Définition du point** et cliquez sur **OK** lorsque vous avez terminé :



La formule n'est pas mise à jour et une icône de mise à jour apparaît en regard de la formule.

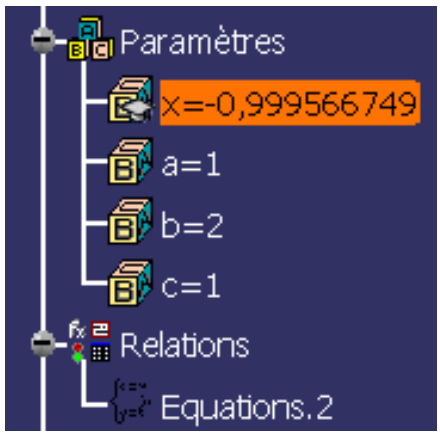
9. Cliquez avec le bouton droit de la souris sur Formula.1 et sélectionnez la commande **Mise à jour locale**. La formule et Longueur.2 sont mis à jour.

Mise à jour d'une équation

Dans cet exemple, l'utilisateur change 3 paramètres, a, b et c, avant de résoudre l'équation. S'il utilise le mode **Synchrone**, une erreur s'affiche lors de la modification des valeurs des paramètres puisque la mise à jour est lancée à chaque modification. Pour garantir la stabilité de l'équation, la mise à jour doit être lancée après la modification de 3 paramètres. L'utilisateur doit donc sélectionner le mode **Mise à jour de la relation lors de la commande de mise à jour globale**.

1. Ouvrez le fichier KwrUpdatinganEquation.CATPart. L'image suivante s'affiche :

   Ce document contient 4 paramètres. x est un paramètre évalué par une équation basée sur les 3 autres paramètres (a, b et c).

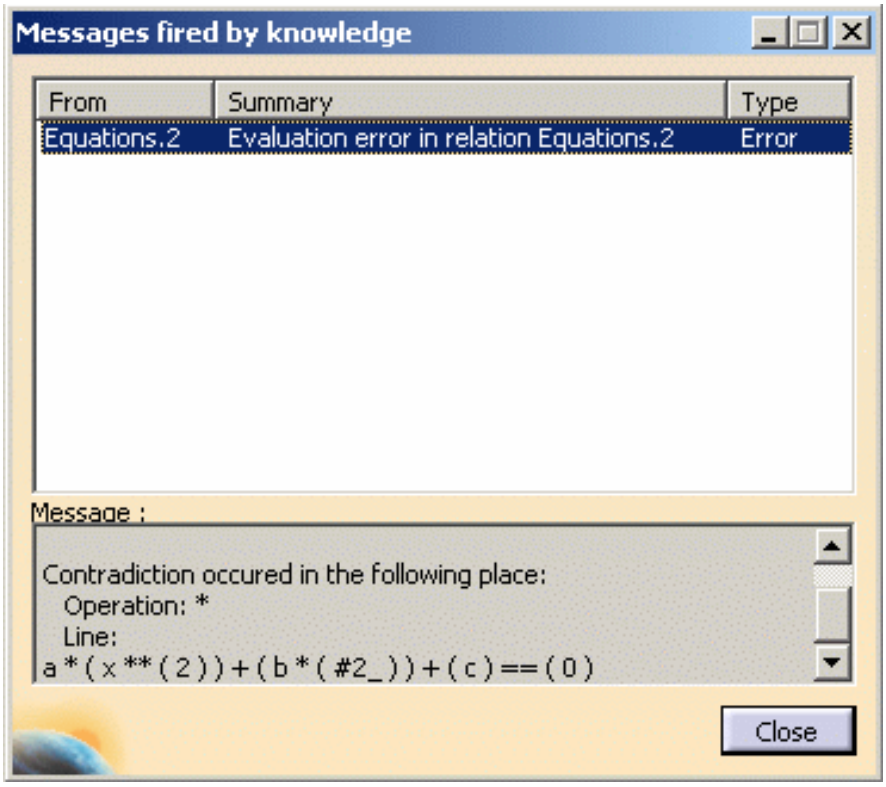


2. Développez le noeud Relations (si besoin est), cliquez avec le bouton droit de la souris sur Equations.2 et sélectionnez la commande **Propriétés**.

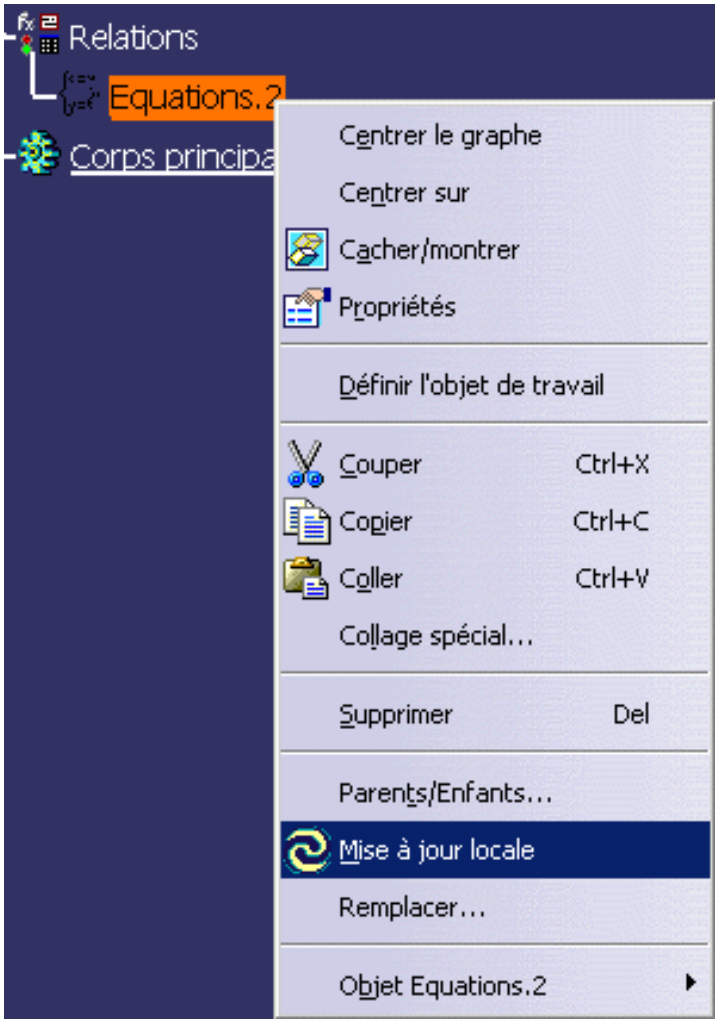
3. Dans la fenêtre Propriétés, activez l'option **Synchrone** et désactivez l'option **Mise à jour de la relation lors de la commande de mise à jour globale**. Cliquez sur **OK** pour valider.

4. Cliquez deux fois dans l'arbre de spécifications. La fenêtre Editer paramètre s'affiche.

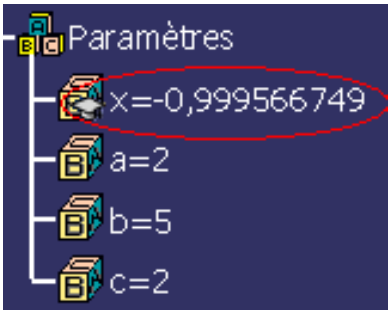
Entrez 2 dans le champ de valeur. Un message indique que l'équation ne peut pas être résolue et cliquez sur **OK**.



5. Cliquez sur **Fermer**.
6. Cliquez avec le bouton droit de la souris sur Equations.2 et sélectionnez la commande **Propriétés**.
7. Dans la fenêtre **Propriétés**, désactivez l'option **Synchrone** et activez l'option **Mise à jour de la relation lors de la commande de mise à jour globale**. Cliquez sur **OK** pour valider.
8. Cliquez deux fois dans l'arbre de spécifications. La fenêtre Editer paramètre s'affiche. Entrez 2 dans le champ de valeur. Cliquez sur **OK**.
9. Cliquez deux fois sur b dans l'arbre de spécifications. La fenêtre Editer paramètre s'affiche. Entrez 5 dans le champ de valeur. Cliquez sur **OK**.
10. Cliquez deux fois sur c dans l'arbre de spécifications. La fenêtre Editer paramètre s'affiche. Entrez 2 dans le champ de valeur. Cliquez sur **OK**.
11. Cliquez avec le bouton droit de la souris sur l'équation dans l'arbre de spécifications et sélectionnez la commande **Mise à jour locale**.



L'équation est mise à jour, ainsi que la valeur x. Pour que l'équation soit résolue, l'option **Mise à jour de la relation lors de la commande de mise à jour globale** doit être activée et la relation doit être intégrée à la mise à jour globale ou mise à jour manuellement à l'aide de la commande Mise à jour locale.



Utilisation de la fonction Action

Dans cette tâche, vous apprendrez à utiliser une action.

L'exemple ci-dessous se divise en trois étapes principales :

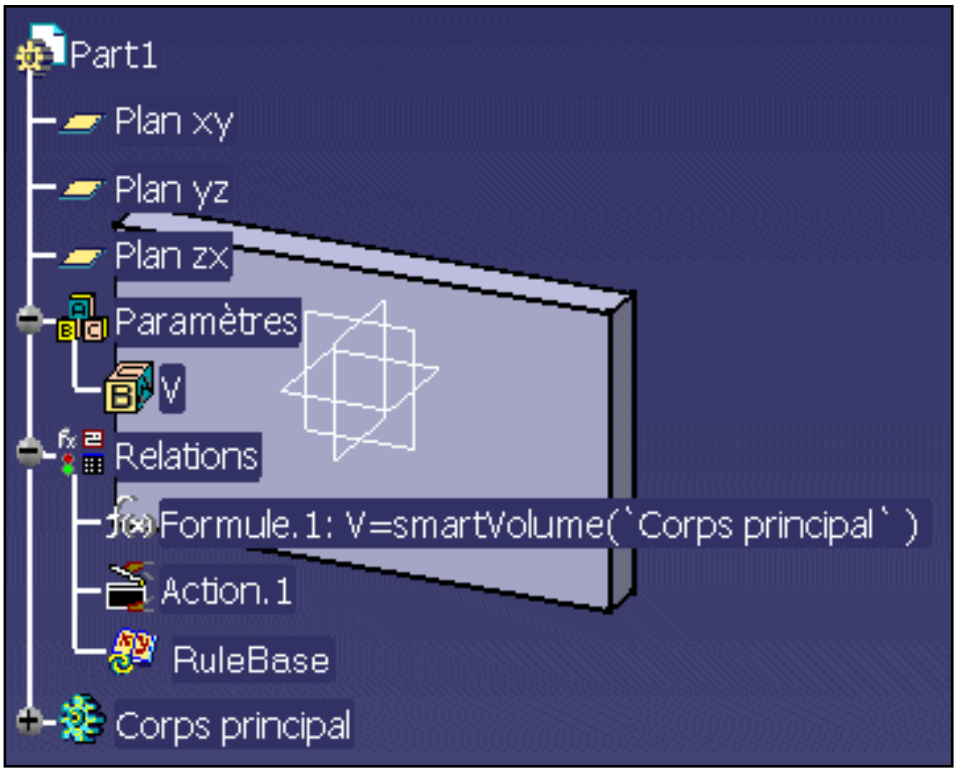
- D'abord, vous créez une extrusion contenant une action.
- Ensuite, vous stockez cette action dans un catalogue.
- Enfin, vous importez dans un autre programme CATPart l'action stockée dans le catalogue.

Pour réaliser cet exemple, il est fortement recommandé de connaître l'atelier Part Design.

1. Ouvrez l'atelier Part Design et créez une extrusion ou ouvrez le fichier KwrAction.CATPart.
2. Créez un paramètre de type Volume et affectez-lui une formule.

   Cliquez sur l'icône  , sélectionnez **Volume** dans la liste déroulante, cliquez sur le bouton **Créer paramètre de type**, puis renommez le paramètre (V dans cet exemple).

   - Cliquez sur le bouton **Ajouter formule**. L'éditeur de formules s'affiche.
   - Sous **Dictionnaire**, sélectionnez **Mesures de pièce**, puis cliquez deux fois sur **smartVolume**. Placez le curseur entre les parenthèses et sélectionnez PartBody. Cliquez sur **OK**, **Oui** (dans le cas d'une mise à jour automatique),



puis **OK**.

3. Ouvrez l'atelier Knowledge Advisor et cliquez sur l'icône Action ( ) afin de créer une action. L'éditeur d'action s'affiche. Entrez le script suivant, puis cliquez sur **OK** :

Champ des entrées	B: Body
Editeur	B.Query("Pad","").Compute("+","Solid","smartVolume(x)",V) Message("Volume total des extrusions sous ce corps : #",V)

- L'action créée ci-dessus recherche les extrusions contenues dans le corps sélectionné et calcule le volume de celles-ci.
- Pour plus d'informations sur la recherche et le calcul, cliquez ici.
- Pour afficher le nouveau fichier .CATPart, cliquez ici.
- Notez que les types d'arguments doivent être écrits en anglais.

4. Enregistrez le fichier et stockez la nouvelle action dans un catalogue.

- Dans le menu **Démarrer**, sélectionnez **Infrastructure->Editeur de catalogue**. L'éditeur de catalogue s'affiche.

- Cliquez sur l'icône **Ajout d'une famille**), ou sélectionnez les commandes **Insertion -> Ajout d'une famille...** dans le menu principal afin d'afficher la boîte de dialogue Famille de composants. Indiquez le nom de la famille (dans cet exemple, ComponentFamily.2), puis cliquez sur **OK**.

- Dans le catalogue, cliquez deux fois sur la famille ComponentFamily.2, puis cliquez sur l'icône **Ajout d'un composant** (), ou sélectionnez la commande **Insertion -> Ajout d'un composant...** afin d'afficher la boîte de dialogue Définition de la description.

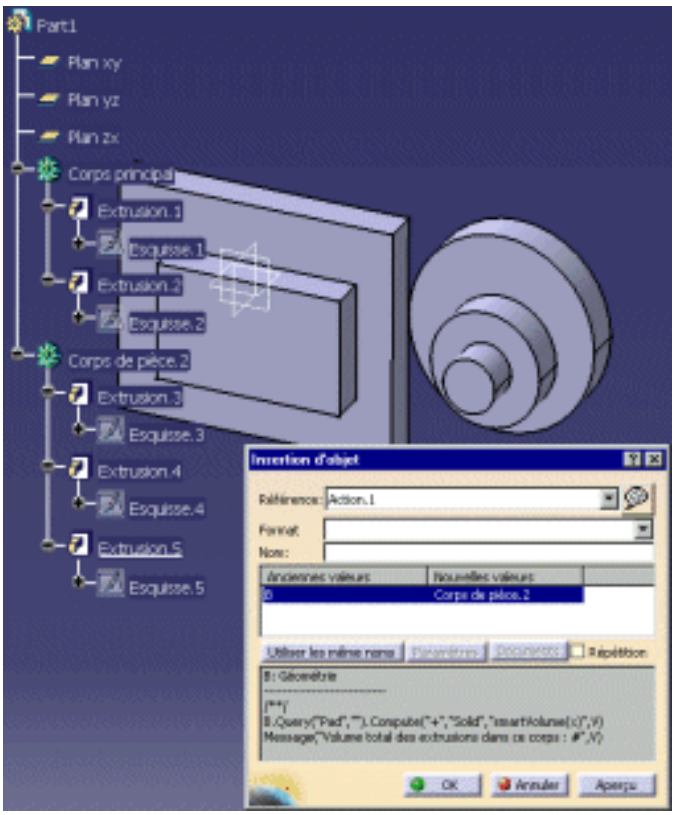
- Cliquez sur le bouton **Sélectionner un feature**. Revenez à la géométrie, sélectionnez l'entité Action.1 dans l'arbre de spécifications, puis cliquez sur **OK**. Enregistrez votre catalogue. L'action contenue dans le fichier .CATPart est maintenant stockée dans le nouveau catalogue.

5. Ouvrez le fichier KwrReceiveAction.CATPart.

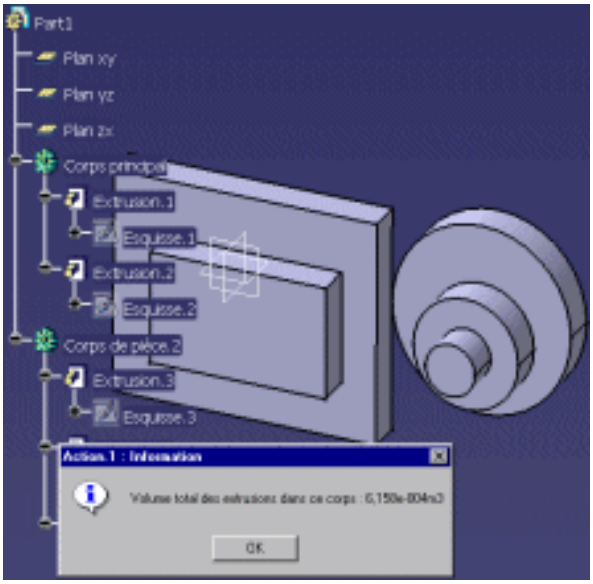
6. Cliquez sur l'icône Catalogue () afin d'importer l'action stockée dans le catalogue.

   Cliquez sur l'icône **Choisir un autre catalogue**), sélectionnez votre catalogue, puis cliquez sur **Ouvrir**. Cliquez deux fois sur ComponentFamily.2 et sur Action.1. La boîte de dialogue **Insertion d'un objet** s'affiche.

(Cliquez sur le graphique ci-contre afin de l'agrandir.)



7. Sélectionnez PartBody : l'action importée affiche le volume des extrusions contenues dans ce corps.



(Cliquez sur le graphique ci-contre afin de l'agrandir.)

8. Sélectionnez Body.2 : l'action importée affiche le volume des extrusions contenues dans ce corps.

Utilisation de la fonction Liste

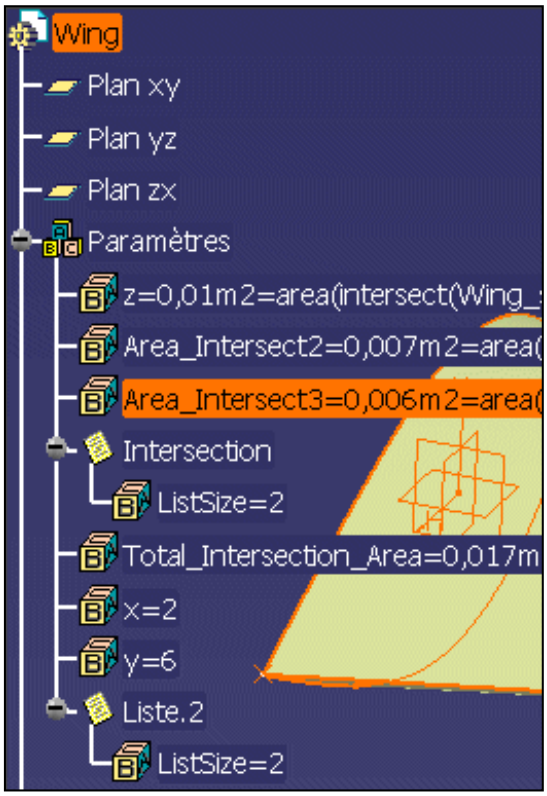
Cliquez sur l'icône Liste pour créer une liste d'entités qui sera située sous le noeud Paramètres dans l'arbre de spécifications. Cliquez sur cette icône pour ouvrir la fenêtre d'édition de la liste.

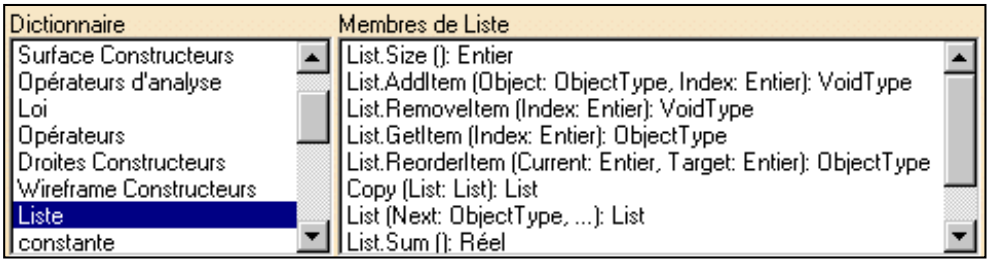
Les entités de liste peuvent être utilisées pour gérer les listes d'objets ou de paramètres. Ces listes peuvent être éditées de manière interactive. La fenêtre Edition de la liste permet de trier les éléments automatiquement et d'indiquer le type des objets autorisés.

La fonction de liste est intégrée au mécanisme de mise à jour ; la taille de la liste est calculée automatiquement. Elle est fournie avec des fonctions permettant de calculer des sommes, des aires, des prix ....

La fonction de liste peut être gérée via le langage pour :

- créer une liste ;
- copier le contenu d'une liste dans une autre ;
- ajouter et supprimer des éléments ;
- obtenir un élément ;
- extraire des valeurs de la liste ;
- déplacer des éléments de la liste vers un autre emplacement.

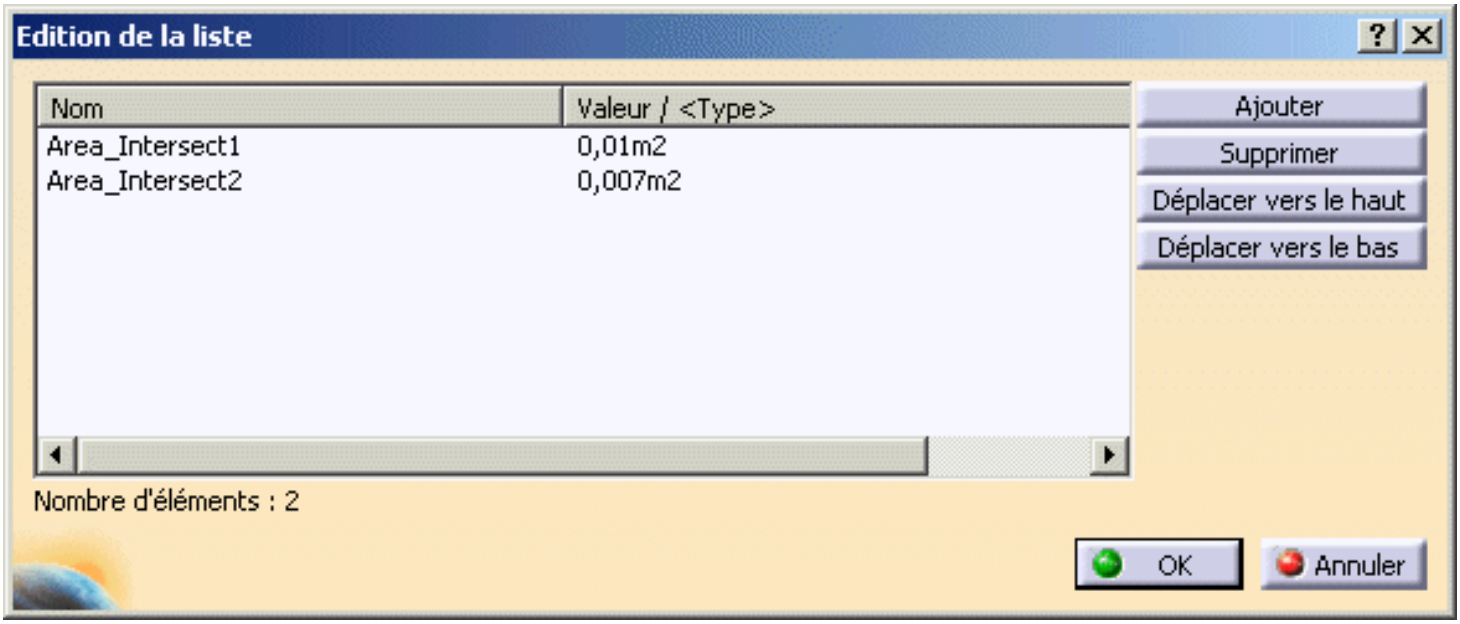




Création d'une liste
Utilisation de la fenêtre Edition de la liste

Utilisation de la fenêtre Edition de la liste

La fenêtre Edition de la liste permet de gérer les objets que vous souhaitez ajouter à la liste que vous êtes en train de créer. La fonctionnalité est accessible en cliquant sur l'icône Liste ( ).



La fenêtre contient quatre boutons différents et est composée de deux colonnes.

Nom	Colonne indiquant le nom de la liste.
Valeur/<Type>	Colonne indiquant la valeur de la liste ou du type associé.
Add	Vous permet d'ajouter à la liste les éléments que vous avez sélectionnés dans l'arbre de spécifications ou dans la géométrie.
Remove	Vous permet de supprimer des éléments de la liste.
Move Up	Vous permet de déplacer vers le haut les éléments de la liste.
Move Down	Permet à l'utilisateur de déplacer vers le bas les éléments de la liste.

Le champ **Nombre d'éléments** affiche le nombre d'éléments que contient la liste.

Il existe 4 types différents de listes :

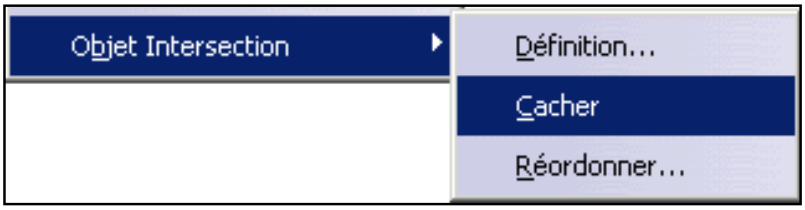
- Listes non visibles ; ces listes sont créées par les utilisateurs CAA et ne peuvent par conséquent pas être modifiées par l'utilisateur en mode interactif (elles ne sont pas affichées).Dans ce cas, les boutons de la boîte de dialogue Liste ne s'affichent pas.

- Listes en lecture seule ; ces listes sont créées par les utilisateurs CAA et ne peuvent par conséquent pas être modifiées par l'utilisateur en mode interactif (elles sont en mode lecture seule).

- Listes en mode lecture/écriture ; ces listes sont créées par les utilisateurs CAA. Elles peuvent donc être éditées mais pas supprimées par l'utilisateur final.

- Listes utilisateur ; ces listes peuvent être éditées et supprimées par l'utilisateur final.

- Comme c'est le cas pour n'importe quel autre paramètre, il est désormais possible de masquer et de réorganiser les listes. Pour ce faire, cliquez avec le bouton droit sur la liste et sélectionnez la commande **Masquer** ou **Réordonner...**.



- Lorsque vous cliquez sur l'icône Liste pour créer une liste, la fenêtre Multi-sélection s'affiche désormais. Pour plus d'informations sur cette fenêtre, reportez-vous au document *Catia Infrastructure - Guide de l'utilisateur*.



Si vous sélectionnez un élément dans la liste, que vous cliquez sur un autre élément dans l'arbre de spécifications ou dans la géométrie et que vous cliquez sur **Ajouter**, alors l'élément de la liste sera remplacé par celui que vous venez d'ajouter.

Création d'une liste

Dans cette tâche, vous apprendrez à utiliser la fonction Liste.
Dans l'exemple ci-dessous, vous allez manipuler une aile de plan à laquelle vous ajouterez des plans et des intersections. Ensuite, vous créerez des paramètres et des formules afin de calculer la surface des intersections, puis une liste permettant de calculer l'aire totale des intersections.

Pour plus d'informations sur la fenêtre Edition de la liste, voir Utilisation de la fenêtre Edition de la liste.
Pour plus d'informations sur la fonction Liste, voir Utilisation de la fonction Liste.

1. Ouvrez le fichier KwrPlaneWing.CATPart.
2. Ouvrez l'atelier Generative Shape Design.
3. Créez trois plans.

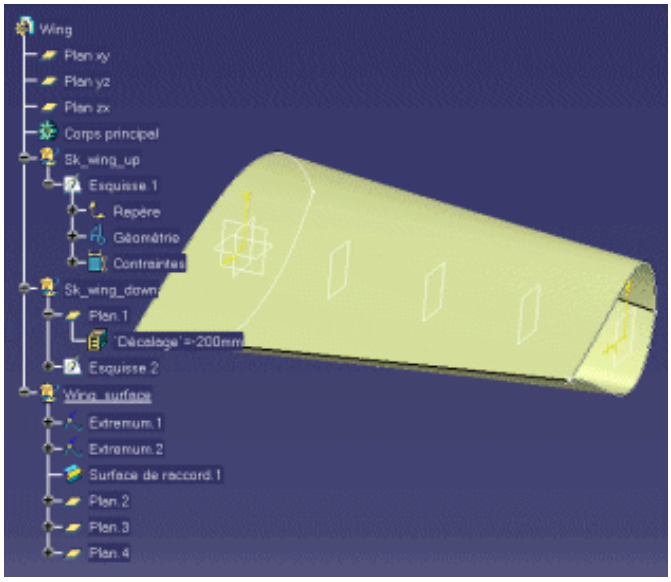
   - Cliquez sur l'icône Plan dans la barre d'outils.
   - a) La boîte de dialogue **Définition du plan** s'affiche.

     b) Dans la zone **Type de plan**, entrez le plan yz (sélectionnez-le dans la géométrie ou dans l'arbre des spécifications) : le plan yz s'affiche dans la zone **Référence**.

     c) Dans le champ **Décalage**, indiquez le décalage requis (par exemple -50 mm). Cliquez sur **OK**.

     d) Répétez deux fois cette opération avec des décalages de -100 et de -150 mm.

     (Cliquez sur le graphique afin de l'agrandir.)



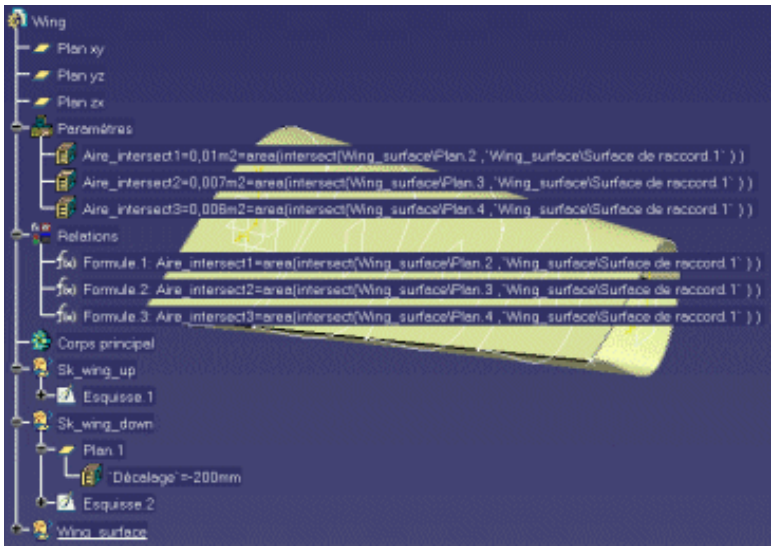
4. Ajoutez des formules pour calculer la surface de l'intersection des plans et de la surface de raccord.

   - Cliquez sur l'icône f(x). L'éditeur de formules s'affiche.
   - Sélectionnez **Aire** dans la liste déroulante et cliquez sur le bouton **Créer paramètre de type**. Remplacez le nom du paramètre par Area_Intersect1, puis cliquez sur le bouton **Ajouter formule**. L'éditeur de formules s'affiche.
   - Dans le **dictionnaire**, sélectionnez **Mesures**, puis cliquez deux fois sur **aire (Surface, ...) : Aire**.
   - Placez le curseur entre les parenthèses, sélectionnez **Wireframe constructeurs** dans le dictionnaire, puis cliquez deux fois sur **intersection (Surface, Surface) : Courbe**.

- Placez le curseur devant la virgule et sélectionnez Plane.2 dans l'arbre de spécifications (ou dans la géométrie), puis sélectionnez le composant Blend.1 dans l'arbre de spécifications.Cliquez sur **OK**.

- Répétez les étapes ci-dessus pour Area_Intersect2 et Area_Intersect3 en sélectionnant Plane.3 et Plane.4.

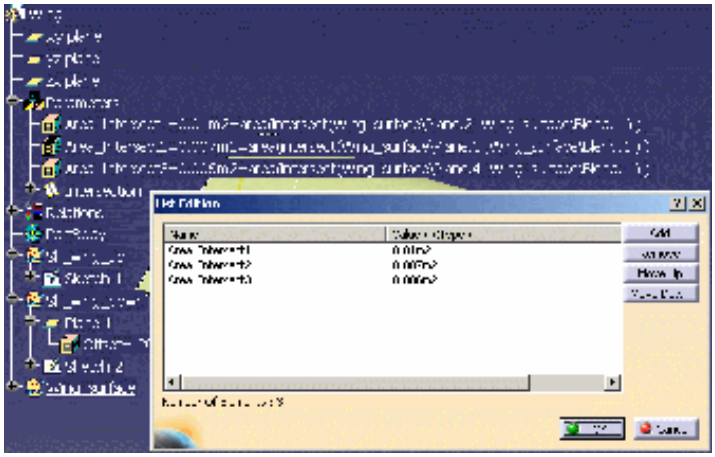
(Cliquez sur le graphique afin de l'agrandir.)



5. Ouvrez l'atelier Knowledge Advisor, puis cliquez sur l'icône Liste ( ). La fenêtre Edition de la liste s'affiche.

6. Sélectionnez les trois paramètres situés sous le noeud Paramètres dans l'arbre de spécifications, puis cliquez sur le bouton **Ajouter**. Cliquez sur **OK**. Renommez par exemple List.1 par Intersections.

(Cliquez sur le graphique afin de l'agrandir.)

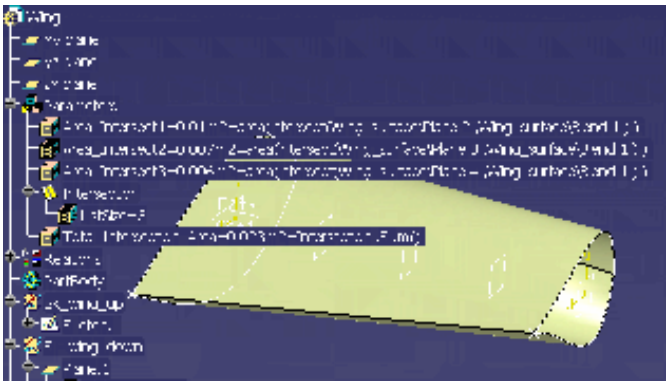


7. Cliquez sur **OK**.La liste est ajoutée aux paramètres.

8. Ajoutez une formule qui calculera l'aire des intersections des trois plans et de la surface de raccord.

   - Cliquez sur l'icône f(x). Sélectionnez **Aire** dans la liste déroulante, cliquez sur le bouton **Créer paramètre de type**, renommez le paramètre Total_Intersection_Areas, puis cliquez sur le bouton **Ajouter formule**. L'éditeur de formules s'affiche.

- Cliquez sur la **liste** (appelée Intersections dans cet exemple) dans l'arbre de spécifications. Le nom de la liste s'affiche dans l'éditeur. Sous Dictionnaire, sélectionnez **Liste**, puis cliquez deux fois sur **List.Sum(): Real** dans la zone **Membres de la liste**. Cliquez deux fois sur **OK**.



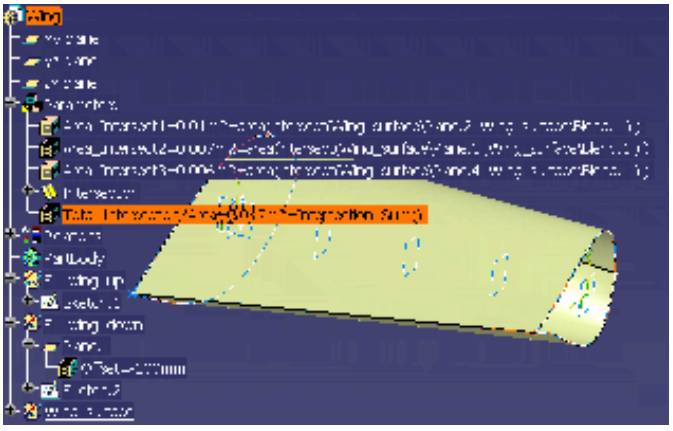
L'aire des intersections des trois plans avec la surface de raccord est calculée automatiquement.

(Cliquez sur le graphique afin de l'agrandir.)

9. Editez le contenu de la liste et recalculez l'aire totale.

- Cliquez deux fois sur la liste (Intersections) dans l'arbre de spécifications : la fenêtre **Edition de la liste** s'affiche. Sélectionnez Area_Intersect3, cliquez sur le bouton **Supprimer**, puis sur **OK**.

- Cliquez avec le bouton droit sur le paramètre Total_Intersection_Areas, puis sélectionnez **Mise à jour locale**.



L'aire des intersections des deux plans restants avec la surface de raccord est calculée automatiquement.

(Cliquez sur le graphique afin de l'agrandir.)

Cliquez ici pour afficher le résultat de cet exemple.

Utilisation de la fonction Boucle

 Cliquez sur l'icône Boucle pour créer une boucle.

Présentation de la fonction Boucle
Familiarisation avec la fenêtre Edition de boucle
Création d'une boucle
Création d'une copie optimisée contenant une boucle
Utilisation du langage de script
Référence

Présentation de la fonction Boucle

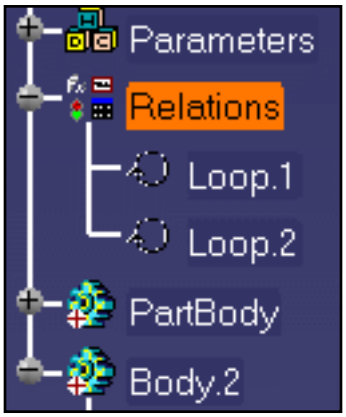
Notez que vous devez disposer d'une licence KWA pour exécuter les boucles.

Les boucles utilisent le langage Generative Knowledge pour piloter la création, la modification et la suppression d'un ensemble d'entités. Cette fonctionnalité permet d'effectuer les opérations suivantes :

- sélectionner des entrées dans la définition de la boucle ;
- définir plusieurs contextes dans l'action de la boucle ;
- inclure la boucle dans une copie optimisée.

La fonctionnalité est accessible en cliquant sur l'icône **Boucle** ( ).

- Une boucle est stockée dans le modèle résultant en tant qu'entité à part entière. Toute modification apportée à leurs spécifications entraînera la modification attendue dans le modèle.
- Une boucle peut être instanciée via une copie optimisée, simplifiant ainsi l'utilisation de manière significative.

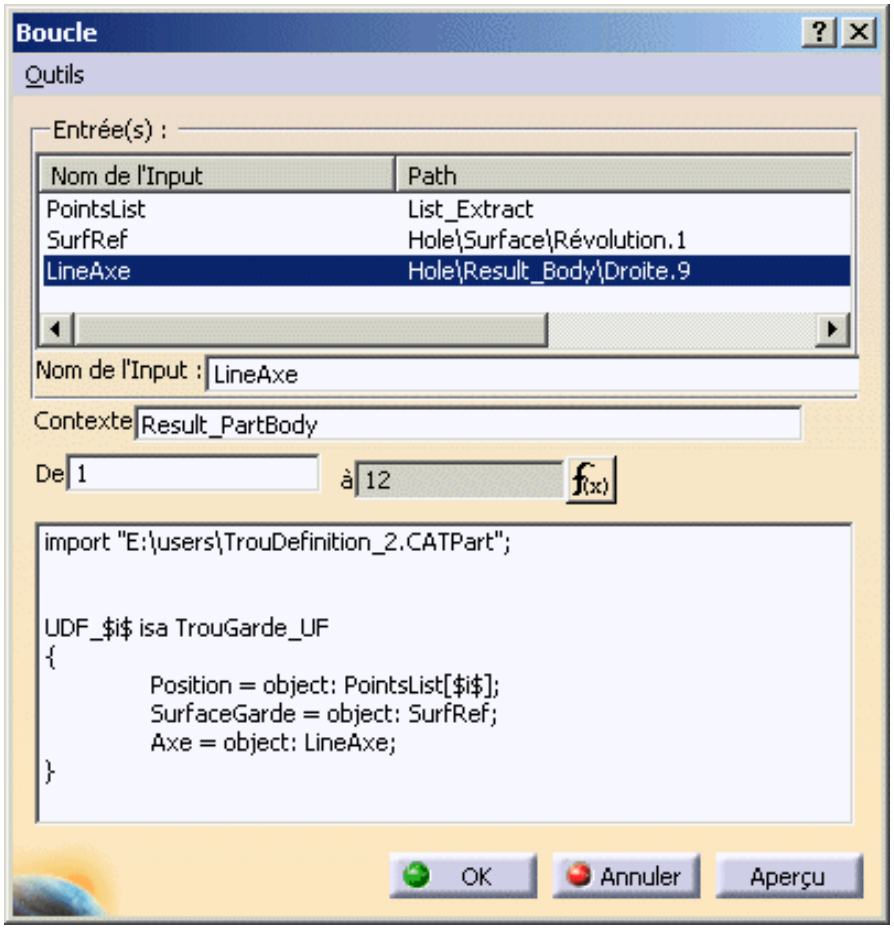


Familiarisation avec la fenêtre Edition de la boucle et les menus

- Fenêtre Edition de la boucle
- Menu Outils de la boucle
- Menu contextuel de la boucle

Fenêtre Edition de la boucle

La fenêtre Edition de la boucle s'affiche lorsque vous cliquez sur l'icône Boucle ( ) dans la barre d'outils **Objets de contrôle**.



Entrée(s)

Ce champ permet de sélectionner les entités que l'utilisateur souhaite utiliser dans l'arbre de spécifications ou dans la géométrie. Les entités sélectionnées sont celles qui seront utilisées dans le corps de la boucle.

Pour désélectionner des éléments dans la liste des entrées, cliquez dessus dans l'arbre de spécifications ou dans la géométrie.

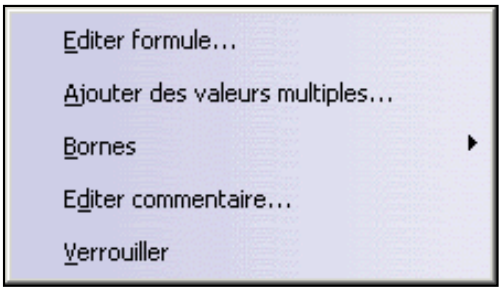
Le champ **Nom de l'entrée** permet de renommer les entrées sélectionnées. Dans ce cas, ce nom sera utilisé dans le corps de la boucle.

Contexte

Ce champ permet de définir le contexte d'application de la boucle. Il peut s'agir de n'importe quelle entité V5. Pour sélectionner le contexte, cliquez sur le champ Contexte, puis sur l'élément correspondant dans l'arbre de spécifications.

Itérateurs

Les champs **De ... à** permettent de définir la fréquence d'exécution de la boucle. Lors de la définition des bornes, vous pouvez cliquer avec le bouton droit de la souris sur les champs **De** et **à** pour accéder au menu contextuel.



- La commande **Editer formule...** permet d'accéder à l'éditeur de formules et de créer une formule qui s'appliquera à l'opération de la boucle. Pour plus d'informations, reportez-vous à la section Création d'une formule.
- La commande **Ajouter des valeurs multiples...** permet d'ajouter plusieurs valeurs. Pour plus d'informations, reportez-vous à la section Passage d'une seule valeur à plusieurs après la création d'un paramètre.
- La commande **Ajouter bornes...** permet d'ajouter une borne.
- La commande **Editer commentaire...** permet d'ajouter un commentaire.
- La commande **Verrouiller** permet de verrouiller ce paramètre. Pour plus d'informations, reportez-vous à la section Verrouillage et déverrouillage d'un paramètre.

Remarques :

- Il s'agit de la première étape dans les champs **De ... à**.
- Les deux limites sont incluses lors de l'exécution de la boucle.

Editeur

L'éditeur permet d'entrer la syntaxe de la boucle. Le langage à utiliser dans cet éditeur est le langage de script. Pour plus d'informations sur la syntaxe à utiliser, voir Utilisation du langage de script.

Menu Outils

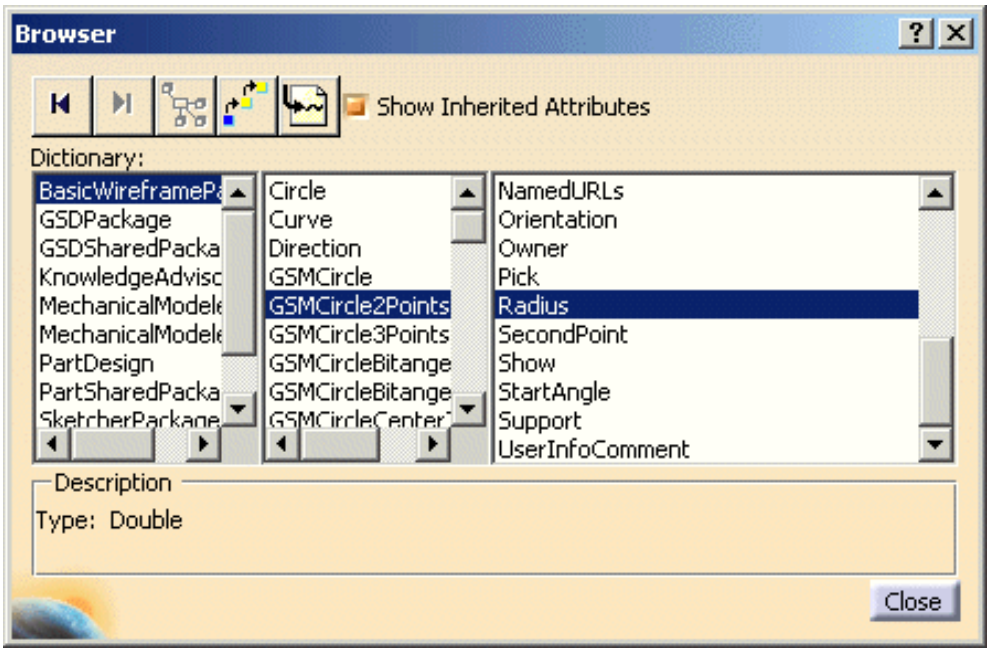
La commande **Outils->Explorateur d'objets...** permet d'accéder à l'explorateur d'objets. Cet explorateur contient les types et les attributs faisant partie de la syntaxe du script.

L'explorateur d'objets vous aide à écrire un script. Il vous permet d'accéder aux mots clés, aux opérateurs et aux attributs d'entité requis lorsque vous utilisez les fonctions de la boucle.

Les modules affichés dans la partie gauche de l'explorateur sont ceux que vous avez sélectionnés via la commande **Outils->Options...**.

Pour ajouter ou supprimer des modules :

1. Sélectionnez la commande **Outils->Options...** pour ouvrir la fenêtre Options, puis sélectionnez **Général->Paramètres et mesure**, puis cliquez sur l'onglet Langage.
2. Dans le champ **Langage** de l'onglet **Connaissance**, cliquez sur **Chargement des bibliothèques d'extension du langage**, puis sélectionnez les bibliothèques.



Dans cette fenêtre, vous pouvez manipuler la liste d'objets pris en charge par le script à l'aide de leurs attributs...

- Dans la partie gauche de l'explorateur sont affichés les modules disponibles.
- La liste des objets appartenant à chaque catégorie est affichée dans la partie centrale.
- Dans la partie droite sont affichés les attributs qui permettent de manipuler ces objets (le cas échéant) .

... et d'écrire des corps de boucle (voir l'exemple ci-dessous) :

```
Circle0 isa GSMCircle
      {
         CircleType = 0;
         TypeObject isa
GSMCircleCtrRad
         {
         Center = object:

..\..\..\Construction_Body\Point.2;
         Support = object:

..\..\..\Construction_Body\Extrude.1;
        Radius = 180mm;
         }
         StartAngle = -100deg;
         EndAngle = 100deg;
        }
```


Icône ***Précédent***.
Pour revenir à la dernière action exécutée dans l'assistant. N'a aucun effet sur l'éditeur de script.


Icône ***Suivant***.
Pour passer à l'action suivante réalisée dans l'assistant, dans une série d'actions.


Icône ***Type d'attribut***.
Cette icône n'est pas disponible dans cette version du produit.

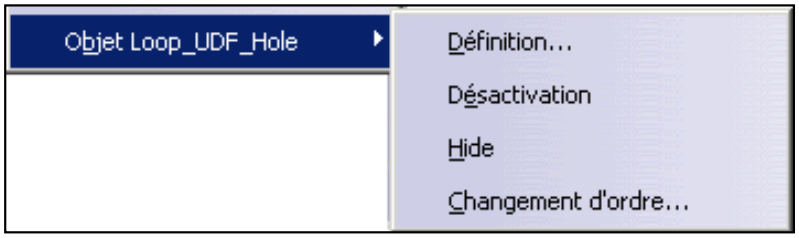

Icône ***Héritage***.
Pour revenir à l'objet racine.


Icône ***Insertion***.
Pour insérer le nom de l'objet dans le script.

Menu contextuel de la boucle

Vous pouvez accéder au menu contextuel de la boucle en cliquant avec le bouton droit de la souris dans l'arbre de spécifications.

- La commande **Définition...** vous permet d'accéder à la fenêtre Edition de la boucle.

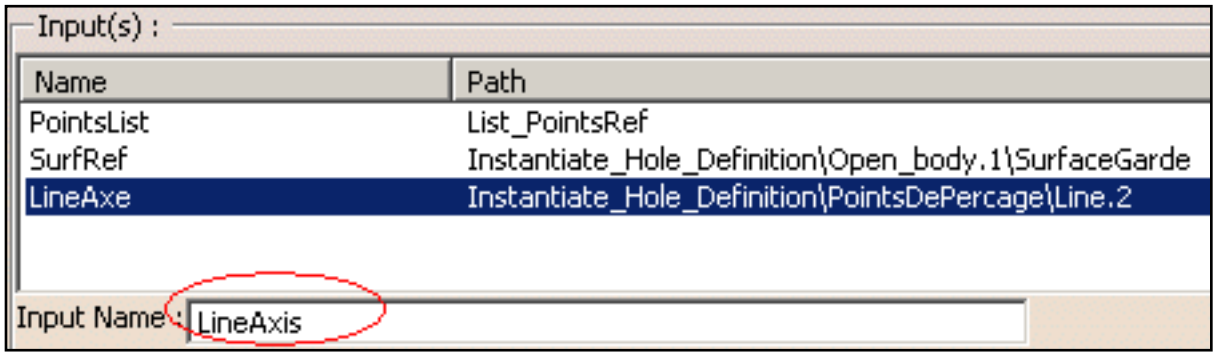


- La commande **Désactivation** vous permet de désactiver la boucle. Dans ce cas, une icône indiquant que la boucle est désactivée s'affiche. Pour l'activer de nouveau, cliquez dessus avec le bouton droit de la souris puis sélectionnez la commande **Activation**.
- La commande **Cachée** vous permet de masquer la boucle. Dans ce cas, elle ne s'affichera pas dans l'arbre de spécifications.
- La commande **Changement d'ordre...** vous permet de changer la position des boucles.

Déclaration des données d'entrée

Les données d'entrée sont les données qui seront utilisées dans le corps de la boucle et qui pourront éventuellement être modifiées lors de l'instanciation de la boucle. Pour les sélectionner, cliquez dessus dans l'arbre de spécifications ou dans la géométrie.

Le champ Nom de l'entrée vous permet de modifier le nom de l'entrée.



Notez que l'entrée peut constituer une liste composée de chaînes. Elle permet d'instancier des entités dont les types sont définis par les éléments de la liste. Pour obtenir un exemple, voir les fichiers suivants :

- KwrPartUDFs.CATPart : Ce fichier contient les 4 définitions des gabarits.
- KwrInstantiateUDFs.CATPart : Ce fichier contient la boucle qui instancie les entités. Les entrées de la boucle sont divisées en deux listes. La première liste contient des paramètres de type chaîne (les noms des gabarits à instancier). La seconde liste contient des paramètres de type Longueur (pour affecter une valeur aux paramètres publiés des gabarits : PosX).

Définition du contexte

Pour créer une boucle, vous devez définir le contexte, c'est-à-dire l'objet (PartBody, Geometrical set, Pad, le noeud Relations, Parameters ou n'importe quelle autre entité) qui contiendra les éléments crées par la boucle. Il existe trois manières différentes de définir le contexte.

Utilisation du champ Contexte

Pour définir le contexte de la boucle, vous pouvez utiliser le champ Contexte de la fenêtre d'édition d'une boucle.
Pour ce faire, cliquez sur le champ Contexte, puis sélectionnez un objet dans l'arbre de spécifications. Dans l'illustration ci-contre, une extrusion a été sélectionnée dans l'arbre de spécifications.



Utilisation d'un document existant

Il est possible d'utiliser un document .CATPart ou .CATProduct existant.

Utilisation du mot clé context

Lors de la création d'éléments devant être placés dans différents corps, vous pouvez modifier le contexte préalablement défini dans le champ Contexte et utiliser le mot clé **Context** pour définir de nouveaux contextes dans le corps de la boucle.

```
Pa1_$i$ = 12mm;
Pt1_$i$ isa GSMPoint{PointType = 0;TypeObject isa GSMPointCoord{ X  = 20mm;}}

context ListeContext[2]
Pa2_$i$ = 1mm;
Pt2_$i$ isa GSMPoint{PointType = 0;TypeObject isa GSMPointCoord{ Y= 15mm;}}


context ListeContext[3]
Pa3_$i$ = 3mm;
Pt3_$i$ isa GSMPoint{PointType = 0;TypeObject isa GSMPointCoord{ Z= InputHauteur;}}
```

Le script d'actions ne doit pas commencer par le mot clé context car le premier contexte est défini dans le champ Contexte.

Exemple : KwrLoopMulticontext.CATPart (pour déclencher la boucle, activez-la.)

Utilisation du langage de script

Présentation du langage de script

Le langage de script permet de générer des entités V5 de manière déclarative. Il permet d'effectuer les opérations suivantes :

- décrire des objets à l'aide d'un langage de script très simple ;
- utiliser des entités géométriques 3D (esquisses, pièces, etc.) ;.
- utiliser des paramètres sur des entités comprenant des *formules* ;
- utiliser des positionnements relatifs & des contraintes d'orientation ;
- générer des modèles V5 correspondants (entités, documents, gabarits, etc.) ;
- entrer le corps des boucles dans la fenêtre Edition de la boucle de l'atelier Knowledge Advisor.

Structure du script d'actions
Propriétés d'objet
Opérateurs
Mots clés
Variables
Commentaires
Limites
Utilisation des commandes Insérer...

Structure du script d'actions

Un script d'actions est écrit au format texte et s'organise en blocs représentant des ensembles d'instructions liés.
Un bloc contient une instruction destinée à créer un objet, suivie d'un ensemble d'instructions entre accolades (**{ }**). Les blocs d'instructions peuvent être imbriqués et celui dont le nombre d'accolades est le plus élevé dans le script correspond aux instructions de création du document.

Un document se compose d'une hiérarchie comprenant des objets, leurs propriétés et les entités qui s'y rattachent. Un script d'actions reflète cette hiérarchie.

Exemple

```
UDF_$i$ isa Hole_UDF 1
{
        Position = object: PointsList[$i$];
2       SurfaceGarde = object: SurfRef;
        Axe = object: LineAxe;
        Garde = 3mm;
}
```

Dans le script ci-contre, les entrées et les paramètres publiés (**2**) du gabarit instancié "Hole_UDF" (**1**) sont imbriqués entre accolades {}.

Propriétés d'objet

- Par défaut, un objet est créé avec certaines valeurs de propriété.Ces propriétés sont définies ou redéfinies entre les accolades juste après la déclaration d'objet (mot clé isa).
- Sauf indication contraire, les unités sont des unités IS.
- Lorsque vous définissez les propriétés, le point-virgule ; est un caractère de terminaison (voir l'exemple ci-après). Les propriétés peuvent être des attributs objet (1), des attributs nécessaires à la définition d'un type affiché dans l'explorateur d'objets (2) ou des objets regroupés (3).

```
UDF_$i$ isa Hole_UDF
{
        Position = object: PointsList[$i$];
        SurfaceGarde = object: SurfRef;
        Axe = object: LineAxe;
        Garde = 3mm;
}
```

1

Dans le script ci-dessus, les propriétés correspondent aux entrées d'un gabarit instancié.

```
Line.1 isa GSMLine
{
    LineType = 0;
    TypeObject isa GSMLinePtPt
    {
     FirstPoint = object: Point.1;
     SecondPoint = object: Point.2;
     Length1 = 500mm; \\ optional
     Length2 = 435mm; \\ optional
    }
   }
```

2

Dans le script ci-dessus, les propriétés correspondent aux attributs requis pour la création d'une droite point à point.

```
MyBody isa OpenBodyFeature
{
  Box isa Pad
       {
        }
}
```

3

Dans le script ci-dessus, l'objet Pad est regroupé sous l'objet OpenBodyFeature.

Mots clés



- Mot clé isa
- Mot clé context
- Mot clé from
- Mot clé import
- Mot clé publish

Mot clé isa

Définition

Permet de créer un objet d'un type donné ou une instance d'un objet.

Syntaxe

- **nom_objet isa type_objet**

ou

- **nom_objet isa nom_instance**

où :

- nom_objet est le nom de l'objet à créer.
- type_objet est le type de l'objet à créer.
- nom_instance est le nom de l'instance d'objet à créer.

Exemple

```
import "E:\GPS.CATPart";
myGps isa CATPart
      {
      myPart isa Part
            {
               PartB isa BodyFeature
                     {
                       S0 isa Sketch.0 {} //Nom de l'instance
                       pad0 isa Pad("S0") //Type de l'objet
                      }
             }
      }
```

Mot clé context

Définition

Permet de définir dans quelle partie de l'arbre de spécifications l'objet sera créé. Le mot clé context peut être utilisé dans 2 cas différents :

- Il peut indiquer le document à utiliser. Dans ce cas, les guillemets ("...") sont utilisés.

```
context "Mypart.CATPart"
MyPart isa Part { }
```

- Il peut faire référence à un objet contenu dans le document. Dans ce cas, le chemin d'accès doit être indiqué (entre `...`).

```
context `My.CATPart\MyPart\PartBody`
CC isa Cylinder { }
```

Syntaxe

- **context "Mypart.CATPart"**

ou

- **context `My.CATPart\MyPart\PartBody`**

Mot clé from

Définition

Permet de copier un document à partir d'un document existant sans avoir à conserver de lien.

Syntaxe

nom_document isa *type_document* from *chemin_fichier*

où :

type_document désigne le document CATProduct, CATPart ou un modèle.
chemin_fichier désigne le chemin d'accès absolu du document de départ.

Pour entrer un chemin d'accès au fichier, vous pouvez :

utiliser la commande Chemin d'accès au fichier du menu contextuel.

Exemple

Reportez-vous à la section Définition du contexte.

Mot clé import

Définition

Désigne un fichier document (.CATPart ou .CATProduct) contenant des définitions à réutiliser ou à redéfinir dans le nouveau document. Le document à créer peut accéder à toutes les entités et les valeurs contenues dans le fichier importé.

L'importation d'un document est :

- intéressante lorsque vous souhaitez extraire un ensemble cohérent de définitions à partir d'un document existant (par exemple pour la définition d'un gabarit).
- obligatoire lorsque vous devez créer une entité à partir d'une esquisse (le langage de script ne permet pas la spécification d'une esquisse).

Syntaxe

Il existe deux manières d'indiquer le chemin du fichier :

- **import "File path";** : Indique le chemin d'accès au fichier à importer : `import "E:\users\kwecx\Models\PartImport.CATPart" ;`. Vous devrez entrer le chemin entre guillemets et terminer l'instruction d'importation par un point virgule (;).
- **import "File Name";** : Indique uniquement le nom du fichier à importer si ce fichier est situé dans le même répertoire que le document contenant la boucle : `import "PartImport.CATPart";`. Remarques :

- Vous devrez entrer le nom du document entre guillemets et terminer l'instruction d'importation par un point virgule (;).
- Le fichier à importer doit résider dans le même répertoire que celui du document contenant la boucle.
- Le document contenant la boucle doit être enregistré.

Pour indiquer un fichier à importer, vous pouvez utiliser la commande Chemin d'accès au fichier à partir du menu contextuel. Lorsque vous sélectionnez cette commande, une fenêtre de sélection de fichier s'affiche. Seuls les guillemets sont automatiquement inclus (le point-virgule ne l'est pas).

Exemple

```
import "E:\users\mei\R12\PES\kwr\loop\Hole_Definition.CATPart";

UDF_$i$ isa Hole_UDF
{
        Position = object: PointsList[$i$];
        SurfaceGarde = object: SurfRef;
        Axe = object: LineAxe;
        Garde = 3mm;
}
```

Mot clé publish

Définition

Permet d'affecter un nom à un objet qui sera utilisé dans le script.

Syntaxe

publish "!xxx" **as** yyy ;

où :

- xxx est le nom de l'objet à publier. Il est recommandé d'utiliser le menu contextuel pour sélectionner cet objet.
- yyy est le nom que vous voulez affecter à cet objet.

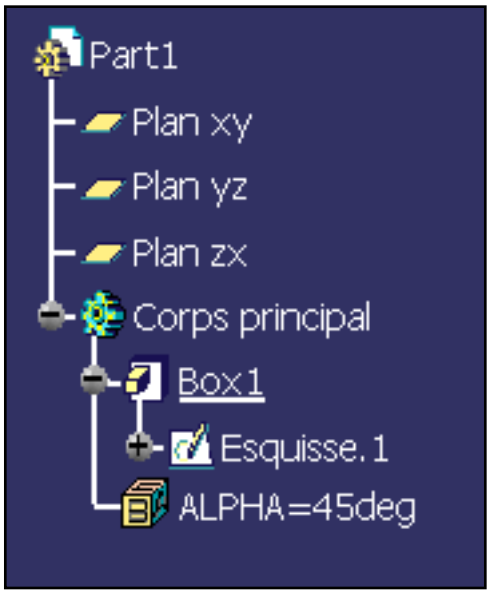
Exemple

```
Publi isa CATProduct
{
   Publi isa Product
   {
       P isa Product
       {
          P1 isa Part
          {
             PartBody isa Feature
             {
                Pa isa Pad{}
             }
          }
       publish "Publi/P/!Selection_RSur:(Face:(Brp:(Pa;2);None:());Pa)" as mypadface; /*publie
       la face d'une extrusion sous le nom "mypadface"*/
       }
       Q isa Product
       {
         Q1 isa Part
         {
            PartBody isa Feature
            {
               Cy isa Cylinder{}
            }
         }
        publish "Publi/Q/!Selection_RSur:(Face:(Brp:(Cy;2);None:());Cy)" as mycylinderface;
        //publie la face d'un cylindre sous le nom "mycylinderface"
        }
        assembly constraints: contact("P\toto","Q\tutu");
    }
}
```

Variables

Les variables sont déclarées de manière explicite dans le script. Elles apparaissent en tant que paramètres dans l'arbre de spécifications.

```
ALPHA = 45 deg;
```



Contrairement à la plupart des langages de script, l'étendue des valeurs d'une variable ne dépend pas réellement de l'emplacement où vous la déclarez. Quel que soit l'emplacement dans le script d'actions, vous pouvez accéder à une variable à l'aide des opérateurs ..\.. et ?. Une fois l'exécution du script terminée, la variable déclarée dans votre script existe toujours en tant que paramètre du document.

Opérateurs

Opérateurs arithmétiques

+ Opérateur d'addition (permet aussi de concaténer des chaînes)

- Opérateur de soustraction

***** Opérateur de multiplication

/ Opérateur de division

( ) Parenthèses (utilisées pour grouper les opérandes dans les expressions)

= Opérateur d'affectation

? (Point d'interrogation dans les formules)

Définition

Dans une formule, spécifie que la valeur de paramètre à appliquer est celle du premier paramètre détecté lors de l'analyse de l'arbre de spécifications depuis la formule jusqu'au sommet de l'arbre de spécifications.

Modèle

KwrLoopRelativePath.CATPart

(Chemin d'accès relatif dans les formules)

Définition

Permet de définir l'endroit où la valeur d'un paramètre utilisé comme argument dans une formule doit être lue.Un seul opérateur .. termine le bloc d'instructions dans lequel la formule est définie. La valeur du paramètre appliquée dans la formule est alors celle définie pour l'entité parente.

Modèle

KwrLoopRelativePath.CATPart

Utilisation des commandes Insérer...

Les commandes décrites dans cette section sont celles auxquelles vous pouvez accéder via l'éditeur de boucle. Il suffit pour cela de cliquer avec le bouton droit de la souris dans la fenêtre de l'éditeur.

Lorsque vous créez une boucle contenant le chemin d'une entité présente dans l'arbre de spécifications, il est vivement recommandé d'utiliser la commande **Insérer élément** pour extraire le nom interne de l'élément.

- Utilisation de la commande Insérer axe
- Utilisation de la commande Insérer arête
- Utilisation de la commande Insérer surface
- Utilisation de la commande Insérer élément
- Utilisation de la commande Chemin d'accès au fichier

Commande 'Insérer axe'

Dans cette tâche, vous apprendrez à créer un chanfrein à l'aide de la commande **Insérer axe**. Cette commande permet de copier le nom générique d'un axe en mode interactif et de l'insérer dans le script au lieu de le taper.

1. Cliquez sur l'icône **Boucle** () et entrez la valeur 1 dans le champ **à**.
2. Dans l'éditeur de script, entrez le script suivant et cliquez sur **OK**. Une extrusion est créée.

```
myChamferDocument isa CATPart
{
	myPart isa Part
	{
		PartBody isa BodyFeature
		{
			P isa Pad
			{
			}

		}
	}
}
```

3. Dans le menu **Fenêtre**, sélectionnez **Cascade**.
4. En dessous du bloc P isa Pad, ajoutez F isa Chamfer(){}, puis cliquez avec le bouton droit pour ouvrir le menu contextuel. Cliquez sur **Insérer axe**, puis

sélectionnez une arête sur votre surface géométrique. Le script doit apparaître comme suit :

```
myChamferDocument isa CATPart
{
      myPart isa Part
      {
            PartBody isa BodyFeature
            {
                P isa Pad
                {
                }
                F isa
Chamfer("Edge:(Face:(Brp:(P;0:(Brp:(Sketch.1;2)));
                None:());Face:(Brp:(P;0:(Brp:(Sketch.1;3)));None:());
                None:(Limits1:();Limits2:()))"){}
            }
       }


}
```

5. Cliquez sur le bouton **OK**. Le chanfrein est créé.

Commande "Insérer arête"



Dans cette tâche, vous apprendrez à créer un chanfrein à l'aide de la commande Insérer arête. Cette commande permet de copier le nom générique d'une arête en mode interactif et de l'insérer dans le script au lieu de le taper.



1. Cliquez sur l'icône **Boucle** () et entrez la valeur 1 dans le champ **à**.
2. Dans l'éditeur de script, entrez le script suivant et cliquez sur **OK**. Une extrusion est créée.

```
myChamferDocument isa CATPart
{
      myPart isa Part
      {
            PartBody isa BodyFeature
            {
                  P isa Pad
                  {
                  }

            }
      }
}
```

3. En dessous du bloc P isa Pad, ajoutez F isa Chamfer(){}, puis cliquez avec le bouton droit pour ouvrir le menu contextuel. Cliquez sur **Insérer arête**, puis sélectionnez une arête sur votre surface géométrique. Le script doit apparaître comme suit :

```
myChamferDocument isa CATPart
{
      myPart isa Part
      {
            PartBody isa BodyFeature
            {
               P isa Pad
               { }
               F isa
Chamfer("Edge:(Face:(Brp:(P;0:(Brp:(Sketch.1;2)));

None:());Face:(Brp:(P;2);None:());None:(Limits1:();Limits2:()))"){}
            }
      }
}
```

4. Cliquez sur le bouton **Génération**. Le chanfrein est créé.

Commande "Insérer surface"



Dans cette tâche, vous apprendrez à créer une esquisse sur une face existante via la commande **Insérer surface**. Cette commande permet de copier le nom générique d'une surface en mode interactif et de l'insérer dans le script au lieu de le taper.



1. Ouvrez le fichier KwrGetSurface.CATPart.
2. Ouvrez l'atelier Knowledge Advisor, puis cliquez sur l'icône Boucle. Entrez la valeur 1 dans le champ **à**.
3. Entrez le script suivant :

```
import "f:\cube.CATPart";
myFaceDocument isa CATPart
{
      myPart isa Part
      {
            PartBody isa BodyFeature
            {
            P isa Pad{}
            S isa Sketch.1()
```

4. Placez le curseur entre les parenthèses qui se trouvent sur la dernière ligne

du script ci-après, cliquez avec le bouton droit pour ouvrir le menu contextuel et sélectionnez la commande **Insérer surface**.

5. Sélectionnez la face dont vous voulez copier le nom. Le nom complet est inséré à l'endroit où se trouve le curseur. Entrez la fin du script. Dans cet exemple, le script final apparaît comme suit :

```
import "f:\PktGetSurface.CATPart";
myFaceDocument isa CATPart
{
      myPart isa Part
      {
          PartBody isa BodyFeature
          {
          P isa Pad{}
          S isa Sketch.1("Face:(Brp:(P;0:(Brp:(Sketch.1;2)));None:())")
          {
          }
          }
      }
}
```

Commande "Insérer élément"



Dans cette tâche, vous apprendrez à utiliser la commande Insérer élément. Cette commande permet de copier le nom générique d'une surface en mode interactif et de l'insérer dans le script au lieu de le taper. Dans la tâche ci-dessous, vous apprendrez à générer une droite.



1. Ouvrez le fichier KwrGetFeature.CATPart.

2. Cliquez deux fois sur la boucle située en dessous du noeud Relations et insérez le code suivant dans l'éditeur :

```
Line_Pt_Pt isa GSMLine
{
LineType = 0;
TypeObject isa GSMLinePtPt
{
FirstPoint = object: // Utiliser commande Insérer élément pour
sélectionner le premier point (Point.10)
SecondPoint = object: // Utiliser commande Insérer élément pour
sélectionner le second point (Point.7)
}
}
```

3. Placez le curseur après FirstPoint = object: puis sélectionnez la commande **Insérer élément** dans le menu contextuel.
4. Cliquez sur un point dans la géométrie (Point.10 par exemple) et ajoutez un point-virgule (;) à la fin de la ligne.
5. Placez le curseur après SecondPoint = object: puis sélectionnez la commande Insérer élément dans le menu contextuel.
6. Cliquez sur un autre point dans la géométrie (Point.7 par exemple) et ajoutez un point-virgule (;) à la fin de la ligne. Votre script doit se présenter comme suit :

```
Line_Pt_Pt isa GSMLine
{
LineType = 0;
TypeObject isa GSMLinePtPt
{
FirstPoint = object: Open_body.1\Open_body.2\Open_body.3\GSMPoint.10;
SecondPoint = object:Open_body.1\Open_body.2\GSMPoint.4;
}
}
```

7. Cliquez sur **OK**. Une nouvelle droite est générée.

Commande 'Chemin d'accès au fichier'



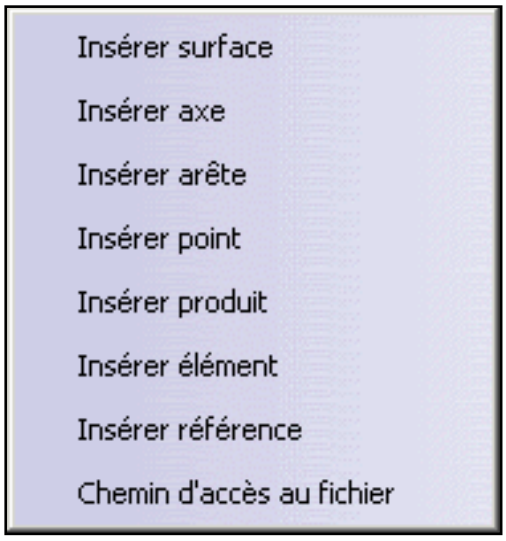
Cette tâche explique comment utiliser la commande Chemin d'accès au fichier. Cette commande correspond à l'une des méthodes utilisées pour indiquer un chemin d'accès dans un script.

Lorsque vous écrivez un script, vous devez spécifier un chemin d'accès lors de l'importation d'un fichier. Pour plus d'informations, voir mot clé import.

1. Ouvrez l'éditeur de script et entrez une instruction nécessitant la spécification d'un chemin de fichier (une importation, comme dans l'exemple ci-après).
2. Placez le curseur à l'endroit où le chemin d'accès doit être spécifié.

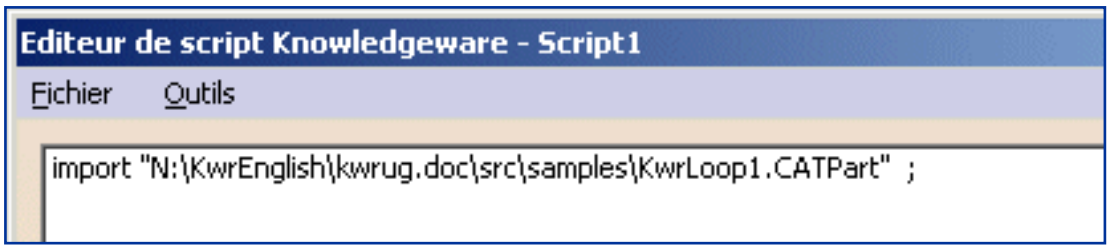


3. Cliquez avec le bouton droit et sélectionnez la commande **Chemin d'accès au fichier** dans le menu contextuel.



4. Dans la boîte de dialogue qui s'affiche, sélectionnez le fichier approprié. Cliquez sur **Ouvrir** pour repasser dans l'éditeur de script.

   Le chemin d'accès absolu est inséré à l'endroit où se trouve le curseur.Vérifiez que l'instruction se termine bien par un point-virgule.



Commentaires

Les commentaires à plusieurs lignes (/* ... */) sont pris en charge. Un commentaire à une seule ligne commence par deux barres obliques (//).

Notez que les caractères DBCS ne sont pas pris en charge en tant que commentaires.

Exemple

```
Sphere1 isa Sphere // Crée une sphère
      {
         // Affecte une valeur à la propriété Radius (rayon)
         Radius = 15.0 ;
      }
```

Limites

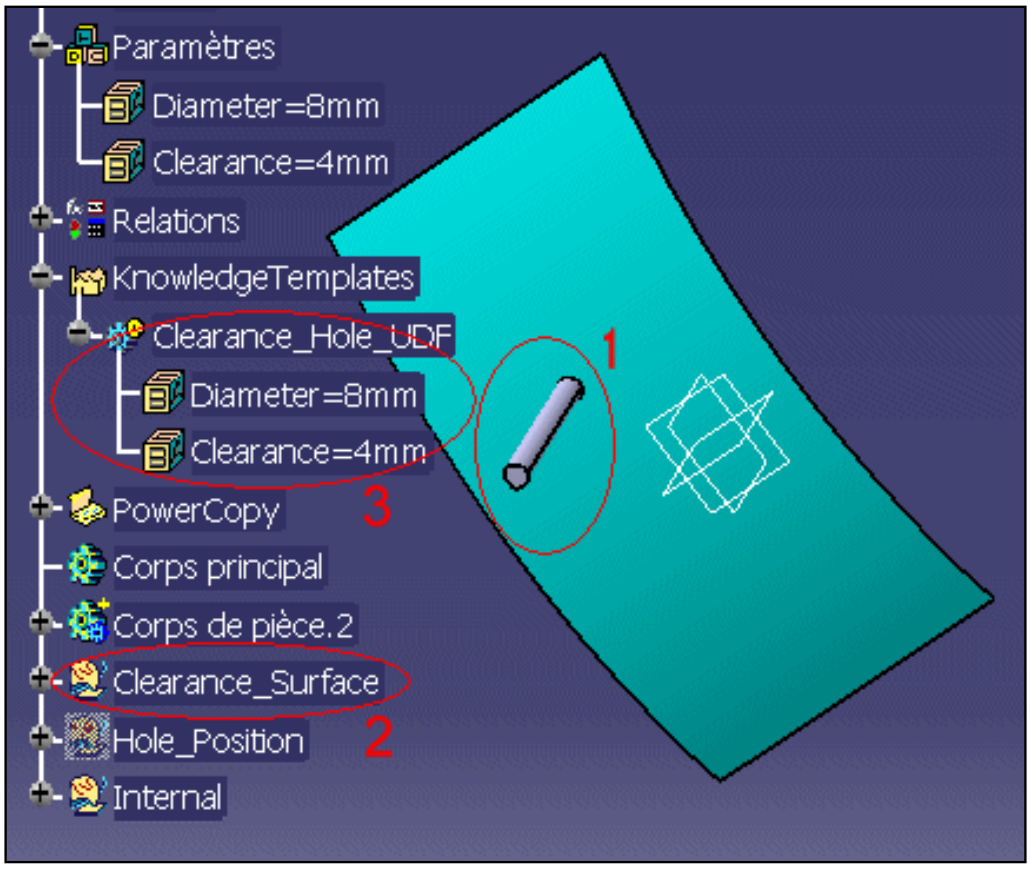


N'oubliez pas qu'il existe quelques restrictions :

- Les instances des entités basées sur esquisse ne peuvent pas être séparées de leur prototype.
- Tout paramètre utilisé comme argument dans une formule doit être précédé du symbole ?.
  La syntaxe X = 2 * Y est incorrecte et doit être remplacée par X = 2 * **?** Y.
- La valeur calculée à partir de la formule apparaît sans unités (sauf si un paramètre défini par une formule n'a pas été initialisé avec les unités qui conviennent).
  Y = 0 kg ;
  Y = 2 * ? X ;
- Une erreur dans le script interrompt la lecture et l'exécution de la boucle.

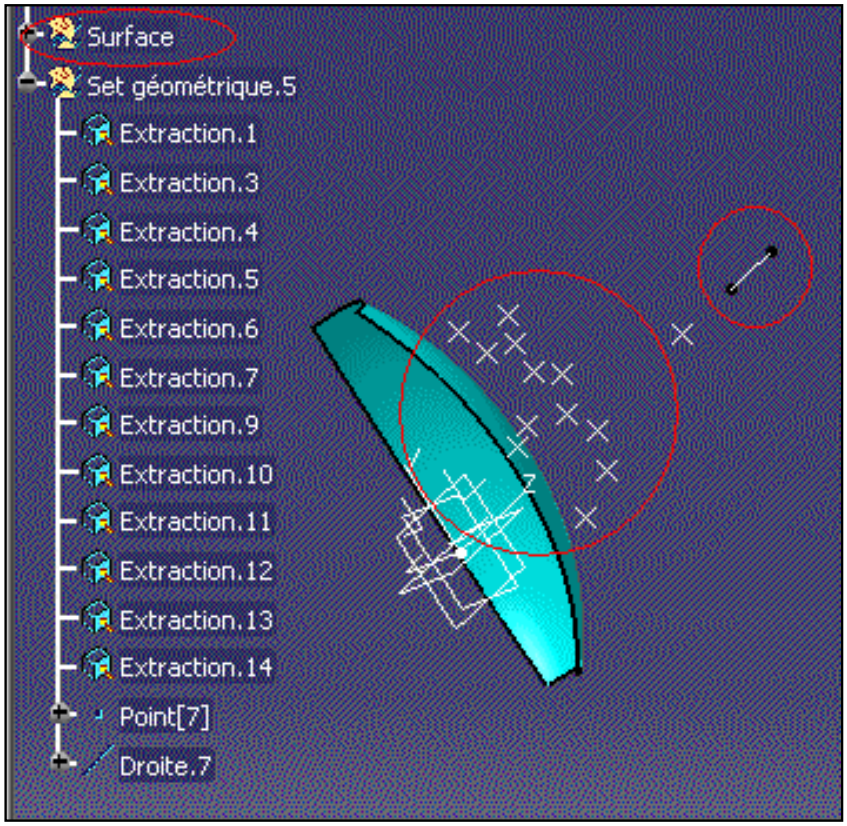
Création d'une boucle

La tâche ci-dessous illustre comment appliquer une boucle à un document existant en mode interactif.



Le document KwrLoop1.CATPart est composé d'une surface (**2**) et d'un solide (**1**) symbolisant un trou. Ce trou est inséré dans un gabarit (UDF) pour être ultérieurement instancié. Le gabarit dispose de 3 entrées différentes (un point, un axe et une surface). Deux paramètres du gabarit sont publiés (**3**).

- Clearance=4mm
- Diameter=8mm



Le document KwrLoop2.CATPart est composé d'une surface (**1**) et de 12 points (**3**) insérés dans une liste. L'élément Line.7 désigne l'axe d'instanciation (**2**).

L'objectif de ce scénario est d'instancier autant de trous qu'il existe de points. Il se divise en plusieurs étapes :

- Vous créez une boucle.
- Vous instanciez le gabarit à partir du fichier .CATPart existant.
- Vous renseignez les entrées nécessaires pour l'instanciation des trous.

Pour créer une boucle, vous devez procéder de la manière suivante :

1. déclarer les données d'entrée ;
2. définir le contexte ;
3. indiquer des itérateurs ;
4. écrire le corps du script d'actions.

Avant de créer une boucle dans un document CATPart, assurez-vous que l'option **Saisie clavier** est **désactivée** dans le champ Référence du produit de l'onglet **Outils->Options->Infrastructure->Product Structure->Product Structure**.

Pour réaliser ce scénario, vous devez utiliser les fichiers suivants :

- KwrLoop1.CATPart
- KwrLoop2.CATPart

1. Ouvrez le document KwrLoop2.CATPart.

Création d'une boucle

2. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier **Knowledge Advisor**.
3. Cliquez sur l'icône **Boucle** () dans la barre **Objets de contrôle**. La fenêtre Edition de la boucle s'affiche.
4. Dans l'arbre de spécifications, sélectionnez les entrées de la boucle.
   - Développez le noeud Paramètres et sélectionnez la liste List_Extract. Dans le champ Nom de l'entrée, entrez le nom de la liste : PointsList.
   - Développez le noeud Surface et sélectionnez l'entité Revolute.1. Dans le champ Nom de l'entrée, entrez le nom de la liste : SurfRef.

- Développez le noeud Result_Body et sélectionnez l'entité Line.9. Dans le champ Nom de l'entrée, entrez le nom de la liste : LineAxis.

Notez que le nom indiqué dans le champ Nom de l'entrée correspond à celui qui sera utilisé pour le corps de la boucle.

5. Sélectionnez le contexte : dans ce scénario, il s'agit de l'entité qui contiendra les trous instanciés.

- Cliquez sur le champ Contexte.
- Cliquez sur Result_PartBody dans l'arbre de spécifications.

6. Indiquez le nombre de trous que vous souhaitez instancier dans la surface.

- Dans le champ De, indiquez la valeur 1.
- Cliquez avec le bouton droit de la souris dans le champ **à** et sélectionnez la commande **Editer formule...**. L'éditeur de formules s'affiche.
- Dans l'arbre de spécifications, cliquez sur ListSize=12. Cliquez sur **OK** lorsque vous avez terminé. La zone répertoriant le nombre de trous instanciés est désormais renseignée par une formule basée sur la liste, c'est-à-dire sur le nombre de points contenus dans cette dernière.

7. Entrez le script d'actions suivant dans l'éditeur.

```
import "E:\users\Loops\KwrLoop1.CATPart"; 1

UDF_$i$ isa Clearance_Hole_UDF 2
{
        Position = object: PointsList[$i$];
        Clearance_Surface = object: SurfRef; 3
        Axis = object: LineAxis;
}
```

- Utilisez le mot clé import pour indiquer le chemin d'accès au fichier contenant le gabarit (UDF) à instancier. Pour indiquer le chemin d'accès au fichier, il est conseillé d'utiliser la commande Chemin d'accès au fichier disponible dans le menu contextuel pour importer le fichier

KwrLoop1.CATPart. (**1**)

- UDF_i correspond au nom qui sera attribué à chaque instance du trou. A chaque itération, i est remplacé par l'itérateur en cours. (**2**)

- Clearance_Hole_UDF correspond au nom affecté au gabarit dans le fichier KwrLoop1.CATPart. (**2**)

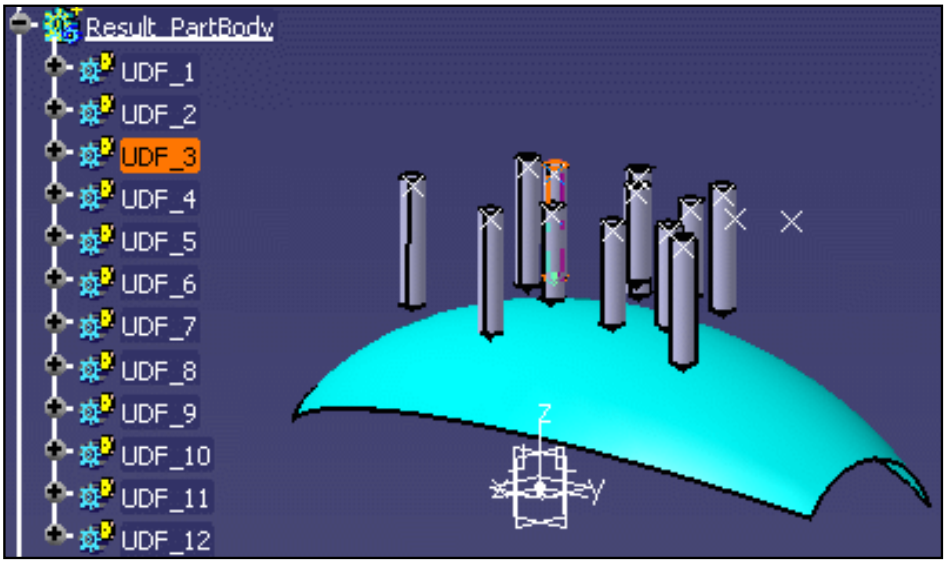
- Position correspond à un point et également à la première entrée à laquelle une valeur doit être affectée lors de l'instanciation des trous. PointsList correspond au nom de la liste. (**3**)

- Clearance_Surface correspond à la seconde entrée nécessaire et définie lors de la création du gabarit et SurfRef correspond à la révolution dans laquelle les trous seront instanciés. (**3**)

- Axis correspond à la troisième entrée nécessaire et définie lors de la création du gabarit et LineAxis correspond à l'entité Line.9, c'est-à-dire à l'axe d'instanciation. (**3**)

Pour plus d'informations sur la syntaxe à utiliser (;, {}, i) dans le corps de la boucle, voir Utilisation du langage de script.

8. Lorsque vous avez terminé, cliquez sur **OK**. Les 12 trous sont instanciés. (Voir illustration ci-dessous.)

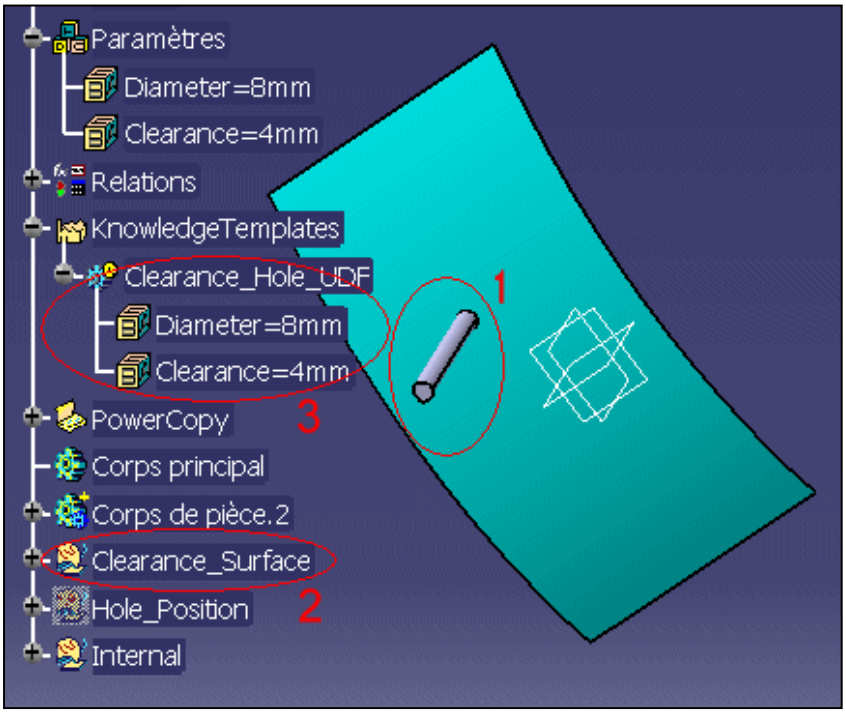

Result_PartBody
UDF_1
UDF_2
UDF_3
UDF_4
UDF_5
UDF_6
UDF_7
UDF_8
UDF_9
UDF_10
UDF_11
UDF_12
Z
Y

Création d'une copie optimisée contenant une boucle

Cette tâche illustre comment appliquer une boucle à un document existant en mode interactif. Dans ce scénario, vous allez faire des trous dans une extrusion. Pour ce faire, vous allez :

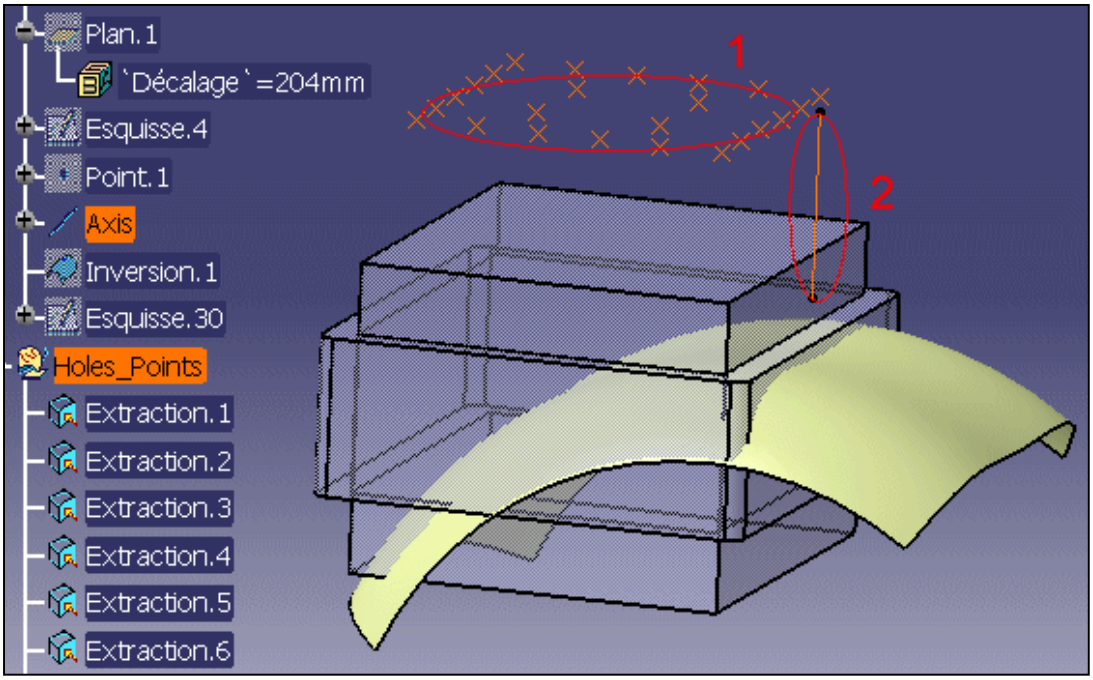
- créer une boucle en mettant en référence les entrées d'un gabarit (UDF) existant utilisé pour faire des trous dans une extrusion ;
- enregistrer la boucle dans une copie optimisée ;
- instancier la copie optimisée dans un document existant et créer les trous.

Pour réaliser ce scénario, vous devez utiliser les fichiers suivants :

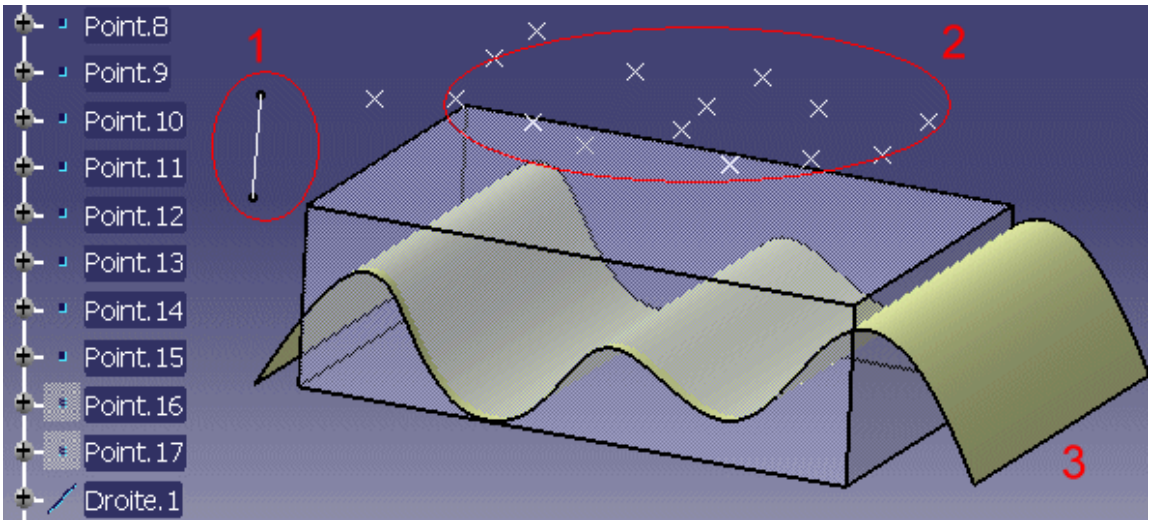


Le document KwrLoop1.CATPart est composé d'une surface (**2**) et d'un solide (**1**) symbolisant un trou. Ce trou est inséré dans un gabarit (UDF) nommé Clearance_Hole_UDF pour une instanciation ultérieure. Le gabarit dispose de 3 entrées différentes (un point, un axe et une surface). Deux paramètres du gabarit sont publiés (**3**) :

- Clearance=4mm
- Diameter=8mm



Le document KwrLoop3.CATPart est composé d'une extrusion, d'une surface et de 24 points (**1**) insérés dans une liste. Line.2 désigne l'axe d'instanciation (**2**). Ce fichier .CATPart contiendra la boucle localisée dans la copie optimisée qui sera instanciée dans le fichier KwrLoop4.CATPart.



Le document KwrLoop4.CATPart est composé d'une extrusion, d'une surface (**3**) et de 17 points (**2**) insérés dans une liste. Line.1 désigne l'axe d'instanciation (**1**). Ce fichier contiendra la boucle instanciée et les trous.

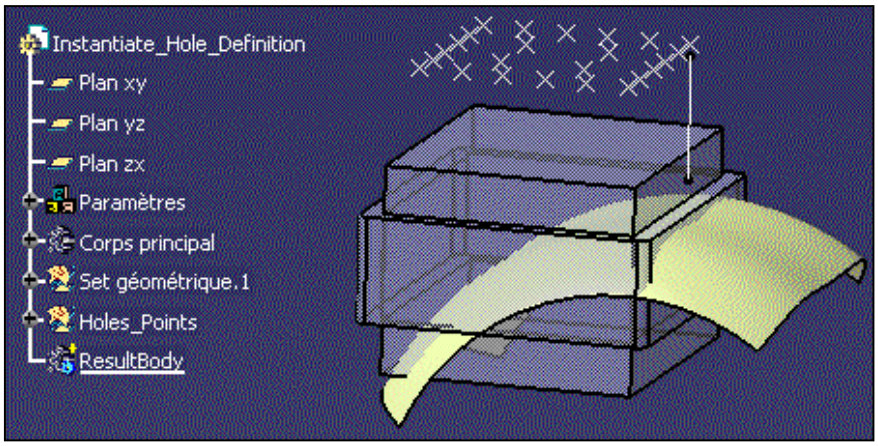
Avant de créer une boucle dans un document CATPart, assurez-vous que l'option **Saisie clavier** est **désactivée** dans le champ Référence du produit de l'onglet **Outils->Options->Infrastructure->Product Structure->Product Structure**.

Pour créer une boucle, vous devez procéder de la manière suivante :

1. déclarer les données d'entrée ;
2. définir le contexte ;
3. indiquer des itérateurs ;
4. écrire le corps du script d'actions.

Création de la boucle en mettant le gabarit en référence

1. Ouvrez le fichier KwrLoop3.CATPart. L'image suivante s'affiche.



2. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier **Knowledge Advisor**.
3. Cliquez sur l'icône **Boucle** ( ) dans la barre **Objets de contrôle**. La fenêtre Edition de la boucle s'affiche.
4. Dans l'arbre de spécifications, sélectionnez les entrées de la boucle.
   - Développez le noeud Paramètres et cliquez sur la liste **List_PointRef**. Dans le champ Nom de l'entrée, entrez le nom de la liste : PointsList.

- Développez le noeud Geometrical Set.1 et sélectionnez l'entité Clearance_Surface. Dans le champ Nom de l'entrée, entrez le nom de l'entité : SurfRef.

- Développez le noeud Holes_Points et sélectionnez l'entité Line.2. Dans le champ Nom de l'entrée, entrez le nom de la droite : LineAxis.

Notez que le nom indiqué dans le champ Nom de l'entrée correspond à celui qui sera utilisé pour le corps de la boucle.

5. Sélectionnez le contexte : dans ce scénario, il s'agit de l'entité qui contiendra les trous instanciés.

- Cliquez sur le champ Contexte.

- Cliquez sur ResultBody dans l'arbre de spécifications.

6. Indiquez le nombre de trous que vous souhaitez instancier dans la surface.

- Dans le champ **De**, indiquez la valeur 1 (1 désignant l'élément Extract.1.).

- Cliquez avec le bouton droit de la souris dans le champ **à** et sélectionnez la commande **Editer formule...**. L'éditeur de formules s'affiche.

- Dans l'arbre de spécifications, cliquez sur ListSize=24. Cliquez sur **OK** lorsque vous avez terminé. La zone répertoriant le nombre de trous instanciés est désormais renseignée par une formule basée sur la liste, c'est-à-dire sur le nombre de points contenus dans cette dernière.

7. Entrez le script d'actions suivant dans l'éditeur.

```
import "E:\users\mei\R12\PES\kwr\loop\test\KwrLoop1.CATPart" ; 1

UDF_$i$ isa Clearance_Hole_UDF  2
{
        Position = object: PointsList[$i$]; 3
        Clearance_Surface = object: SurfRef; 4
        Axis = object: LineAxis; 5
        Clearance = 3mm; 6
}
```

- Utilisez le mot clé import pour indiquer le chemin d'accès au fichier contenant le gabarit (UDF) à instancier (KwrLoop1.CATPart).

- Pour indiquer le chemin d'accès au fichier, il est conseillé d'utiliser la commande Chemin d'accès au fichier disponible dans le menu contextuel pour importer le fichier KwrLoop1.CATPart. (**1**)

- UDF_i correspond au nom qui sera attribué à chaque instance du trou. (**2**)

- Clearance_Hole_UDF correspond au nom affecté au gabarit dans le fichier KwrLoop1.CATPart. (**2**)

- Position correspond à un point et également à la première entrée à laquelle une valeur doit être affectée lors de l'instanciation des trous. PointsList[i] correspond au nom de la liste. [i] correspond au énième élément de la liste. Dans le cas présent, énième équivaut à 24, soit le nombre de trous à instancier (**3**).

- Clearance_Surface correspond à la seconde entrée nécessaire et définie lors de la création du gabarit et SurfRef correspond à la révolution dans laquelle les trous seront instanciés. (**4**)

- Axis correspond à la troisième entrée nécessaire et définie lors de la création du gabarit et LineAxis correspond à l'entité Line.9, c'est-à-dire à l'axe d'instanciation. (**5**)

- Clearance est l'un des paramètres publiés du gabarit. Il est utilisé dans le script d'actions car vous souhaitez que la valeur du paramètre publié soit modifiée. (**6**)

Pour plus d'informations sur la syntaxe à utiliser (;, {}, i) dans le corps de la boucle, reportez-vous à la section Utilisation du langage de script.

8. Lorsque vous avez terminé, cliquez sur **OK**. Les trous sont instanciés (voir graphique ci-dessous).
9. Cliquez sur l'icône **Mise à jour** () pour mettre à jour le document.
10. Cliquez avec le bouton droit de la souris sur la boucle et utilisez la commande **Propriétés** pour renommer la boucle dans Loop_Holes. Cliquez sur **OK** lorsque vous avez terminé.
11. Cliquez avec le bouton droit de la souris sur la boucle dans l'arbre de spécifications (sous le noeud Relations) et sélectionnez la commande **Objet Loop_Holes->Désactiver**.

Enregistrement de la boucle dans une copie optimisée

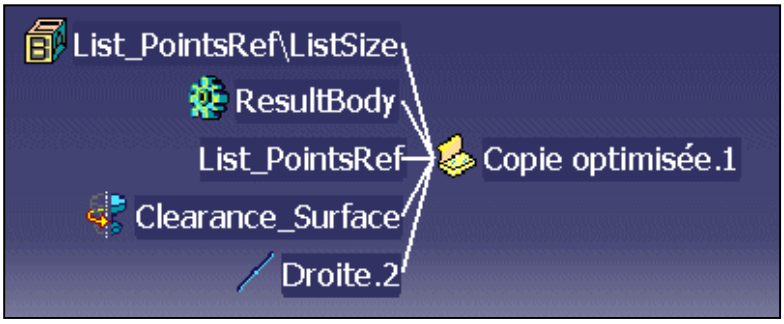
12. Cliquez sur la racine de l'arbre de spécifications et, à partir du menu **Démarrer->Conception mécanique**, accédez à l'atelier **Part Design**.
13. A partir du menu **Insertion->Outils de réplication**, sélectionnez la commande **Création de copie optimisée...**. La fenêtre Définition de copie optimisée s'affiche.
14. Dans l'arbre de spécifications, sélectionnez les éléments constituant la copie optimisée :

- Formula.1

- Loop_Holes

Notez que la copie optimisée nécessitera les entrées suivantes au moment de l'instanciation :

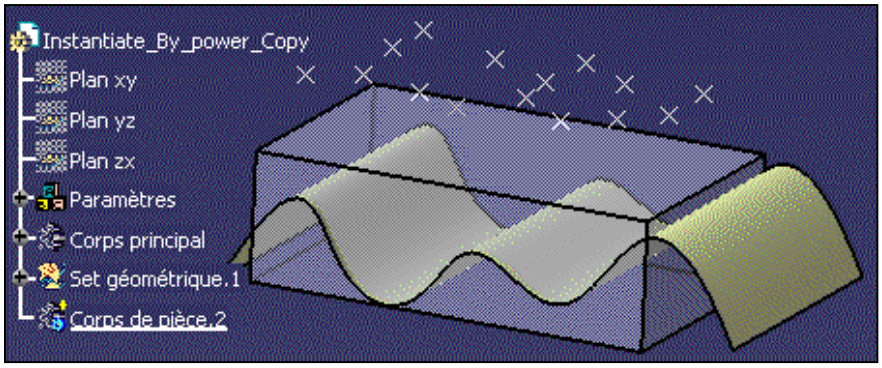
- ListSize
- Line.2
- Clearance_Surface
- List_PointsRef
- ResultBody



15. Cliquez sur **OK** lorsque vous avez terminé. La copie optimisée est ainsi créée et s'affiche sous le noeud PowerCopy dans l'arbre de spécifications.
16. Enregistrez le fichier, puis fermez-le.

Instanciation de la copie optimisée dans un document existant

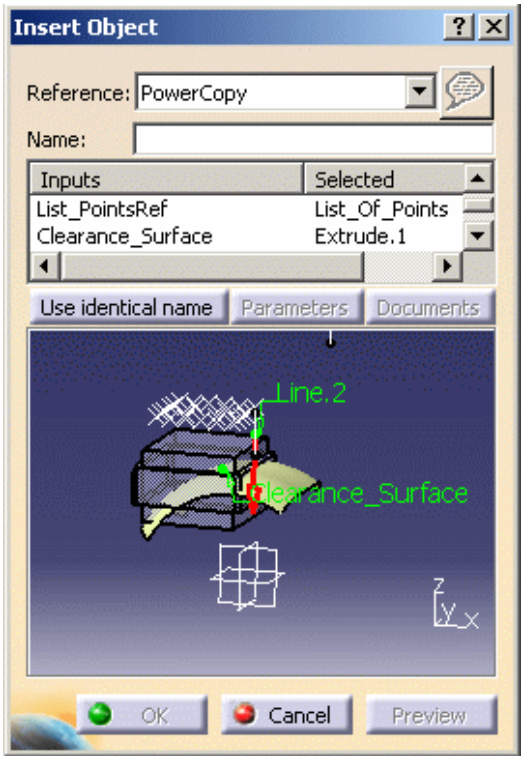
17. Ouvrez le fichier KwrLoop4.CATPart. L'image suivante s'affiche.



18. Depuis le menu **Insertion**, sélectionnez la commande **Instancier à partir d'un document...**.
19. Dans la fenêtre **Sélection de fichier**, sélectionnez le fichier KwrLoop3.CATPart que vous venez d'enregistrer puis cliquez sur **Ouvrir**. La boîte de dialogue Insertion d'objet s'affiche.
20. Indiquez une valeur pour les entrées de la copie optimisée.
    Pour ce faire, cliquez sur les éléments suivants dans l'arbre de spécifications :
    - ListSize=15 pour renseigner l'entrée ListSize.
    - Body.2 pour renseigner l'entrée ResultBody.
    - List_Of_Points pour renseigner l'entrée List_PointsRef.

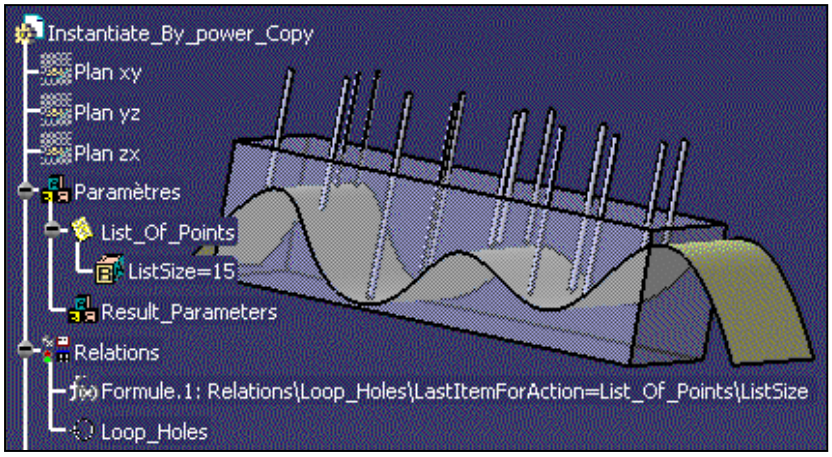
- Extrude.1 (situé sous Geometrical Set.1) pour renseigner l'entrée Clearance_Surface.

- Line.1 (sous Geometrical Set.1) pour renseigner l'entrée Line.2, c'est-à-dire l'axe d'instanciation.



21. Cliquez sur **OK** lorsque vous avez terminé. La boucle et la formule contenues dans la copie optimisée sont instanciées.

22. Pour instancier les trous, vous devez activer la boucle. Pour ce faire, cliquez avec le bouton droit de la souris sur Loop.1 dans l'arbre de spécifications et sélectionnez la commande **Objet Loop.1->Activer**. Les trous sont instanciés.



Résolution d'un ensemble d'équations

Dans cette tâche, vous apprendrez à résoudre un ensemble d'équations à l'aide d'opérateurs et de fonctions de règles de connaissance. Vous pouvez exécuter ce scénario à partir de n'importe quel document.

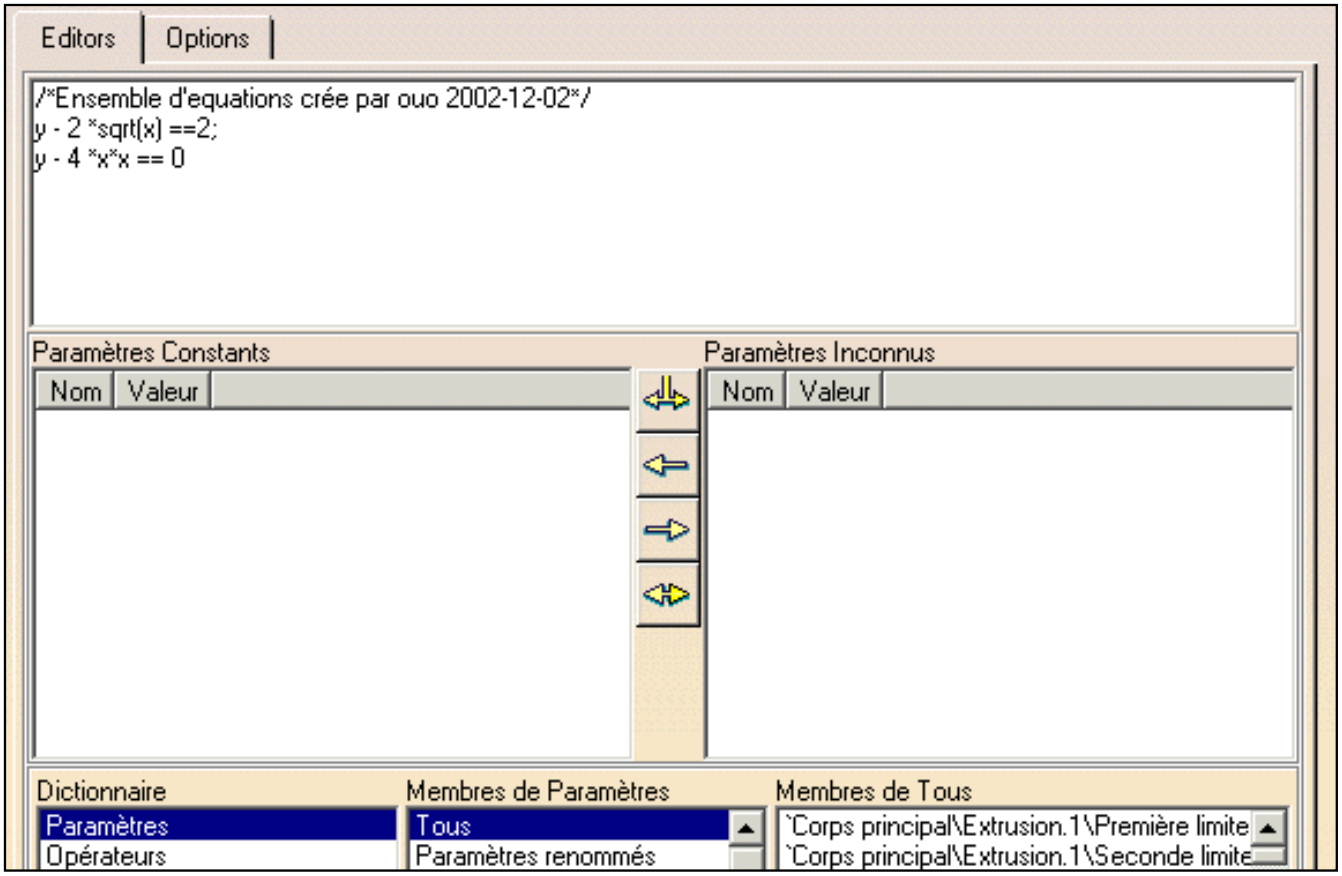
- Dans un ensemble d'équations, le point-virgule (;) est utilisé comme séparateur.
- Notez que les fonctions associées aux ensembles d'équations requièrent le programme Knowledge Advisor.

Notez que vous devez disposer d'une licence KWA pour exécuter les boucles.

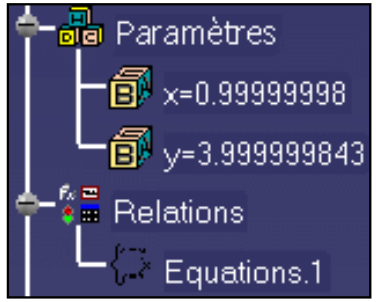
1. Créez deux paramètres de type réel x et y. Les deux paramètres sont destinés à servir de variables dans un ensemble d'équations.
2. Ouvrez l'atelier Knowledge Advisor. Cliquez sur l'icône  . Dans la première boîte de dialogue qui s'affiche, entrez le nom de la relation, un commentaire et une destination. Cliquez ensuite sur **OK**. L'éditeur d'équations s'affiche.
3. Entrez l'ensemble d'équations ci-après dans la zone d'édition :

   ```
   y - 2 *sqrt(x) ==2;
   y - 4 *x*x == 0
   ```

   A présent, l'éditeur ressemble à ce qui suit :



La valeur de chaque paramètre est affichée en premier lieu dans le champ Paramètres Inconnus puis dans l'arbre de spécifications (voir ci-dessous).



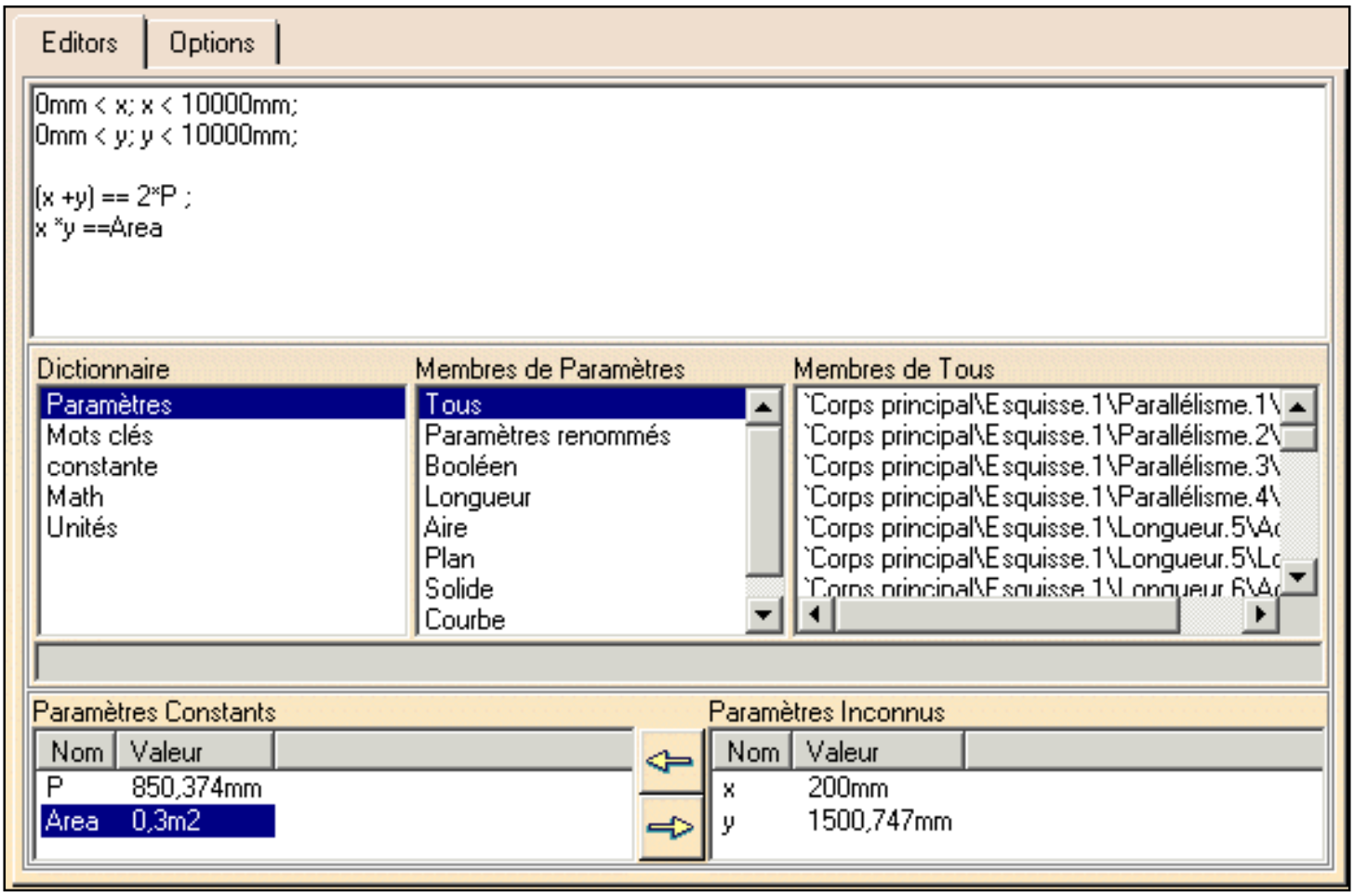
4. Cliquez ici pour ouvrir le fichier de résultats.

Pour plus d'informations sur l'éditeur d'équations, voir Utilisation de l'éditeur d'équations.

Utilisation de l'éditeur d'équations

Afin d'améliorer l'utilisation des fonctions de résolution d'équations, l'éditeur d'équations a été modifié. Il est constitué de deux onglets : l'onglet Editeurs et l'onglet Options.

Onglet Editeurs




La flèche Analyse permet d'identifier les variables de l'ensemble des contraintes. Vous devez cliquer dessus avant de sélectionner les variables d'entrée et de sortie.


La flèche de gauche permet de déplacer les variables de la catégorie Paramètres inconnus vers la catégorie Paramètres constants.

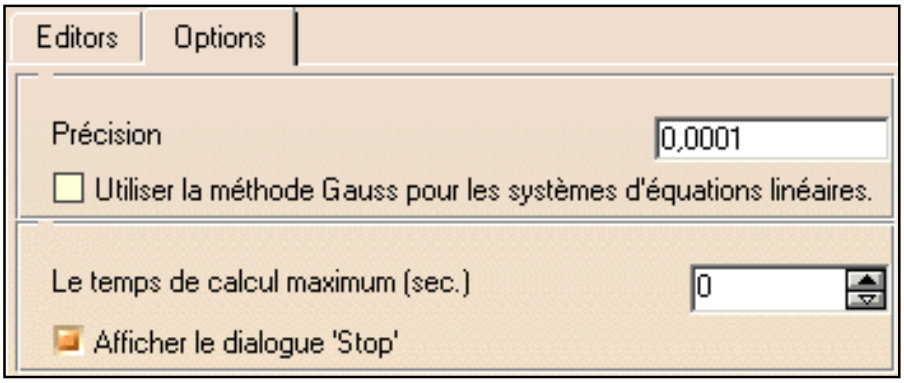

La flèche de droite permet de déplacer les variables de la catégorie Paramètres constants vers la catégorie Paramètres inconnus.


La flèche d'inversion d'entrée-sortie permet de permuter les paramètres constants et les paramètres inconnus.

- Visionneur : permet d'entrer les équations à résoudre.
- Dictionnaire : voir Utilisation de l'éditeur de règles.
- Membres de (paramètres) : voir Utilisation du dictionnaire.
- Membres de : voir Utilisation de l'éditeur de règles.

- Paramètres constants : Les valeurs des paramètres constants sont définies par l'utilisateur et considérées comme constantes par le programme de calcul. Ces valeurs peuvent être modifiées directement dans la colonne Valeur. Pour ce faire, cliquez deux fois (lentement) dans la cellule Valeur.
- Paramètres inconnus: la valeur des paramètres inconnus est calculée lorsque vous cliquez sur le bouton Appliquer.

Onglet Options



Algorithme

- **Précision** : permet de définir la précision des résultats (c'est-à-dire le nombre de chiffres décimaux placés après la virgule).
- **Utiliser la méthode Gauss pour les systèmes d'équations linéaires** : accélère l'opération de résolution lorsque vous travaillez avec des équations linéaires.

Critère de terminaison

- **Temps de calcul maximum (sec.)** : permet d'indiquer le temps de calcul. Si le temps indiqué est égal à 0, le calcul durera jusqu'à ce qu'une solution soit trouvée.
- **Afficher le dialogue 'Stop'** : sélectionné, cet onglet affiche la boîte de dialogue "Stop" permettant d'interrompre le calcul.

Résolution d'un ensemble d'équations

Utilisation du langage Knowledge Advisor

- Ecriture des formules, des règles & des vérifications - Présentation générale
  - Instructions conditionnelles
  - Instruction For
  - Instruction While
- Constantes
- Commentaires
- Variables temporaires
- Unités
- Opérateurs
- Méthodes d'objet
- Messages et macros

Ecriture des formules

Une formule est une instruction réduite à une ligne que vous pouvez créer, soit en tapant directement les commandes requises dans la zone d'édition, soit en sélectionnant les composants à partir de la liste du dictionnaire de l'éditeur. La syntaxe des formules est facile à apprendre et à utiliser.

En règle générale, le point est utilisé comme séparateur entre un nombre et ses décimales. Il n'est pas recommandé d'utiliser une virgule à la place d'un point comme séparateur dans des valeurs réelles destinées à être utilisées dans les relations.
Exemple : L'équation Real1 = 2,1 + 5,4 n'est pas autorisée contrairement à Real1 = Real2 + Real3, quel que soit le séparateur utilisé lors de l'évaluation de Real2 et de Real3.

Ecriture des règles et des vérifications

Les règles et les vérifications sont des instructions à **plusieurs lignes** que vous pouvez créer, soit en tapant directement les commandes requises dans la zone d'édition, soit en sélectionnant les éléments à partir de la liste du dictionnaire de l'éditeur. La syntaxe à utiliser est décrite ci-après. Les fonctions mathématiques, les fonctions trigonométriques et les fonctions de chaîne sont les mêmes que pour les formules.

Instructions conditionnelles

Règles

if ... else ... else if

Exécute de manière conditionnelle un ensemble d'instructions en fonction de la valeur d'une expression. Vous pouvez utiliser des syntaxes contenant des blocs d'instructions :

```
if instructions_condition [else elsestatements ]


ou


if condition
   { instructions}
[else if condition-n
   [ { instructions_elseif } ] ] ...
[else
   [ { elsestatements } ] ]
```

Pour des règles courtes et simples, vous pouvez utiliser le format à une ligne (première syntaxe). Toutefois, le format bloc (deuxième syntaxe) est plus structuré et plus souple que le format à une ligne. En outre, il permet souvent de lire, de gérer et de tester les commandes plus facilement.

Les deux clauses **else** et **else if** sont facultatives. Vous pouvez avoir autant d'instructions **else if** que vous le souhaitez après un bloc **if**, mais aucune ne peut apparaître après la clause **else**. Les blocs d'instructions **if** peuvent être imbriqués, c'est-à-dire contenus les uns dans les autres.

Vérifications

instruction1 **=>** *instruction2* (**if** *instruction1* **then** *instruction2*)
Affiche un message (si le type est Warning ou Information) et devient rouge dans l'Arbre de spécifications chaque fois que l'*instruction2* est incorrecte et que l'*instruction1* est satisfaite.

OK => KO 

KO => KO 

KO => OK 

OK => OK 

Instruction For

Notez que l'instruction For est uniquement disponible pour les scripts Action et Réaction.

Le premier type de boucle est basé sur l'élément d'une liste. Voir la syntaxe ci-contre.

```
For x inside List
{
... Body ...
}
```

où :

- X est un nom de variable (d'un type donné. Il peut représenter un objet ou une valeur).
- List est un nom de variable de type List ou une expression renvoyant une liste.
- X (comme toute autre variable du langage) peut être utilisé dans le corps. Il contient le Nième élément de la liste.

Le corps est exécuté N fois où N correspond au nombre d'éléments de la liste.

Le second type de boucle s'exécute jusqu'à ce qu'une expression ait la valeur "false". Voir la syntaxe ci-contre.

```
For x while predicate
{
... Body ...
}
```

où :

- X est un nom de variable de type entier. Il est incrémenté à la fin de chaque exécution du corps.
- Le prédicat est une expression booléenne. Le corps est exécuté tant que cette expression a la valeur "true". Cette expression est calculée avant le corps.

Notez que le second type peut générer des boucles infinies.

Instruction While

Cette boucle s'exécute jusqu'à ce qu'une expression ait la valeur "false". Voir la syntaxe ci-contre.

```
let i = 1
let x(Point)

for i while
i<=parameter.Size()
{
x = parameter.GetItem(i)
if (x.GetAttributeReal("Y") <
0.04)
x.SetAttributeReal("Y",0.04)
}
```

où :

- i est un nom de variable de type entier. Il est incrémenté à la fin de chaque exécution du corps.
- X est une variable pour les points.

Constantes



Lors de la programmation des règles et des vérifications, *CATIA* peut reconnaître un certain nombre de constantes. Vous pouvez utiliser ces constantes dans une relation à la place des valeurs

proprement dites.

- faux - Une des deux valeurs admises pour un opérateur booléen.
- vrai - Une des deux valeurs admises pour un opérateur booléen.
- PI - **3.14159265358979323846** - Rapport de la circonférence d'un cercle à son diamètre.
- E - Base du logarithme népérien. La valeur de la constante *e* est approximativement **2,718282**.



Commentaires

Les caractères de commentaire **/*** et ***/** sont pris en charge.

```
/* Règle créée par CRE 05/03/99 */
if PartBody\Sketch.1\Radius.3\Radius > 45mm
  {
   LaunchMacroFromFile("Macro1.CATScript")
  }
else
/*
   LaunchMacroFromFile("Macro2.CATScript")
*/
   Message("Aucune macro exécutée")
```

Variables temporaires



Les variables temporaires peuvent être déclarées par le biais du mot clé **let**. Une variable temporaire n'existe plus, en tant que paramètre, une fois la règle exécutée.

```
/*Règle créée par CRE 08/23/99*/
let x = 5 mm
if PartBody\Hole.1\Diameter > x
{
PartBody\Hole.1\Activity = false
}
```

Le type doit être indiqué pour les valeurs non numériques :

```
let S(Surface)
S= split (...,...)
```

Les variables temporaires doivent être déclarées au début de la règle, avant toute spécification

d'instruction conditionnelle.

```
let S1(Surface)
let S2(Surface)
let S3(Surface)


S1 = Split ...
S2 = ...
S3 = ...
```

Unités

Toutes les unités apparaissent dans le dictionnaire.

1. Faites attention à la cohérence des unités utilisées lors de l'écriture d'une règle ou d'une vérification.
2. Les unités s'expriment avec un soulignement au lieu du signe "/" habituel (par exemple N_m2 au lieu de N/m2).

Opérateurs

Opérateurs arithmétiques

+ Opérateur d'addition (permet aussi de concaténer des chaînes)

- Opérateur de soustraction

***** Opérateur de multiplication

/ Opérateur de division

( ) Parenthèses (utilisées pour grouper les opérandes dans les expressions)

= Opérateur d'affectation

** Opérateur d'élévation à une puissance

Opérateurs logiques

et Conjonction logique de deux expressions

ou Disjonction logique de deux expressions

Opérateurs de comparaison

<> Différent de

== Egal à

>= Supérieur ou égal à

<= Inférieur ou égal à

< Inférieur à

> Supérieur à

Méthodes d'objet

Description

Décrit le parent de l'ensemble des entités mécaniques.

Attributs

ID	Owner
Name	NamedURLs
UserInfoComment	

Méthodes

Méthode AbsoluteId	Méthode AttributeType
GetAttributeInteger	SetAttributeBoolean
GetAttributeString	GetAttributeReal
Méthode ID	HasAttribute
Méthode IsOwnedBy	Méthode IsOwnedByString
Méthode IsSupporting	Méthode Name
Méthode Query	SetAttributeInteger
SetAttributeBoolean	SetAttributeReal
SetAttributeString	

Exemple

1. Créez une pièce comprenant plusieurs trous.
2. Ajoutez un paramètre de type réel ("Real.1" par exemple) à l'une des entités d'un trou. Pour ce faire, vous devez utiliser le produit Knowledge Advisor.
3. Créez la règle ci-après :

- List.1 est le nom de la liste concernée par le calcul ;
- PartBody est le corps sur lequel portera la recherche ;
- Hole désigne le type de l'entité.
- x.Diameter>50mm désigne l'expression.

```
/* Cette règle réaffecte la valeur Real.1 au diamètre du trou */
/* dont le paramètre est "Real.1" */
(for all) H:Hole
if H->HasAttribute("Real.1")
H.Diameter = 1mm*(H->GetAttributeReal("Real.1"))
```

Vous pouvez utiliser l'ensemble des méthodes GetAttributexxx de cette manière.

- Ajoutez une ou plusieurs dépouilles à la pièce.
- Vous pouvez écrire la règle ci-dessous :

```
(for all) Dr:Draft
/* Affiche les noms des dépouilles portant le nom PartBody */
```

Voir également : Utilisation du dictionnaire

Attributs



Id
Désigne l'identifiant de l'entité, c'est-à-dire le premier nom qui lui a été attribué lors de sa création avant qu'elle ne soit renommée.

Owner
Désigne l'entité parente.

Name
Désigne le nom de l'entité.

NamedURLs
Décrit l'URL que l'utilisateur peut ajouter à une relation en cliquant sur l'icône Commentaire et URLs dans l'atelier Knowledge Advisor.

UserInfoComment
Décrit le commentaire que l'utilisateur peut ajouter dans la boîte de dialogue Commentaire et URLs lorsque vous ajoutez un URL à une relation dans l'atelier Knowledge Advisor.

Limites

Paramètres et formules

- Le paramètre de sortie (celui qui est renseigné) d'une formule ne peut pas être une publication. La solution consiste à utiliser une règle.
- Si vous voulez modifier une formule présente dans un autre document (autre que l'interface utilisateur active) dans l'éditeur de formules, un message d'avertissement s'affiche et l'éditeur de formules devient indisponible.



Tables de paramétrage

Lorsque vous utilisez des tables de paramétrage au format .xls, la commande **Annuler** risque de ne pas fonctionner correctement.



Règles

- Les entités géométriques renseignées par une règle ne doivent pas être utilisées comme support de construction des sous-éléments.
- Si vous créez une activité de formules générant des règles, vous pouvez rencontrer des problèmes de mise à jour.



Ensembles d'équations

Dans les ensembles d'équations, il n'existe pas de contrôle de la cohérence des unités (comme dans une règle, une vérification ou une formule).

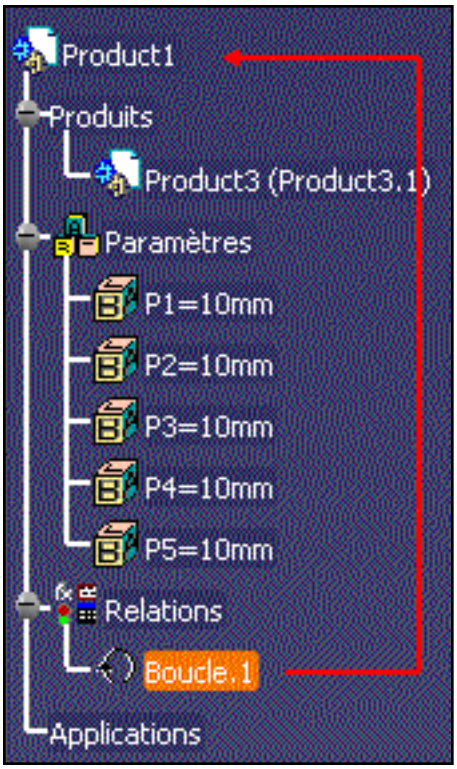


Macros avec des arguments

Il n'est pas possible d'utiliser la fonction InputBox Visual Basic dans la définition des macros avec des arguments sur Unix. Par ailleurs, la méthode SelectElement sur l'objet Selection ne fonctionne pas (sur tous les systèmes d'exploitation).



Boucles



Une boucle peut uniquement créer des éléments dans le document auquel elle appartient. Ainsi, comme le montre le graphique ci-contre, la boucle ne peut pas créer d'éléments dans Product3.

Cette restriction s'applique également aux assemblages composés de pièces différentes. Une boucle située dans une des pièces ne peut pas créer d'éléments dans une autre pièce de l'assemblage.

Conseils utiles

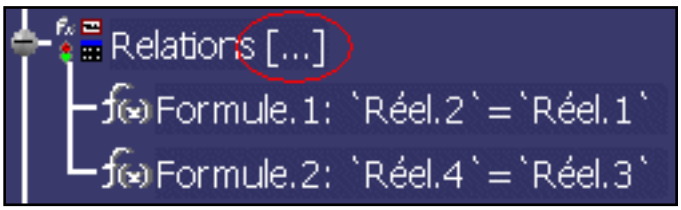
Relations

Lorsque vous utilisez certains objets, vous devez indiquer la cible des formules et des règles qui définissent les paramètres de ces objets. Si vous entrez un paramètre de temps dans une simulation cinématique, par exemple, la relation ne sera pas localisée sous l'ensemble des relations mais dans l'arborescence des mécanismes et des commandes de la simulation.

Masquage des relations

Vous pouvez masquer une relation Knowledge (formule, règle, vérification, etc.) en cliquant avec le bouton droit sur cette relation dans l'arbre de spécifications et en sélectionnant la commande **Cacher**.

- Un indicateur visuel situé au niveau de l'ensemble des relations indique que l'ensemble contient des relations cachées. Notez que cet indicateur n'est pas récurrent.
- Si vous essayez de supprimer un ensemble de relations contenant des relations cachées, un message vous demande si vous souhaitez supprimer l'ensemble de relations qui contient les relations cachées.
  Notez que si vous essayez de supprimer un ensemble de relations contenant un autre ensemble de relations composé de relations cachées, le message s'affiche.



Mises à jour des relations

L'évaluation des relations contenant des mesures peut être intégrée uniquement à la mise à jour de la pièce. Dans un fichier .CATProduct ou .CATProcess, pour créer un paramètre

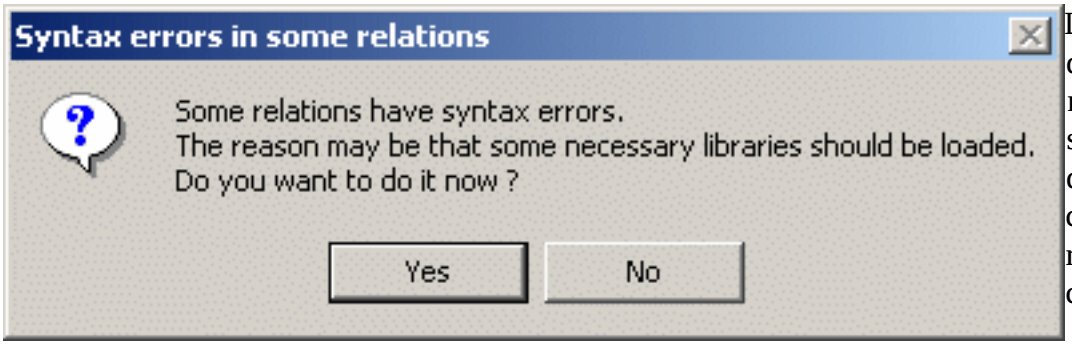
- dont la valeur est le résultat d'une relation contenant des mesures
- mises à jour lors de la modification des entrées de mesure :

1. Créez la relation contenant la mesure au niveau de la pièce.
2. Intégrez l'évaluation de la relation à la mise à jour de la pièce.
3. Au niveau du produit ou du processus, créez une relation définissant le paramètre en utilisant le paramètre de résultat de la relation créée au niveau de la pièce. Pour obtenir un exemple, reportez-vous

au document KwrUpdate.CATProduct.

4. Procédez à une mise à jour locale au niveau des relations.

Erreurs de syntaxe dans un message de relations



Lors du chargement d'un document, le message ci-contre peut s'afficher. Il signifie que pendant le chargement, certaines relations sont apparues comme brisées.

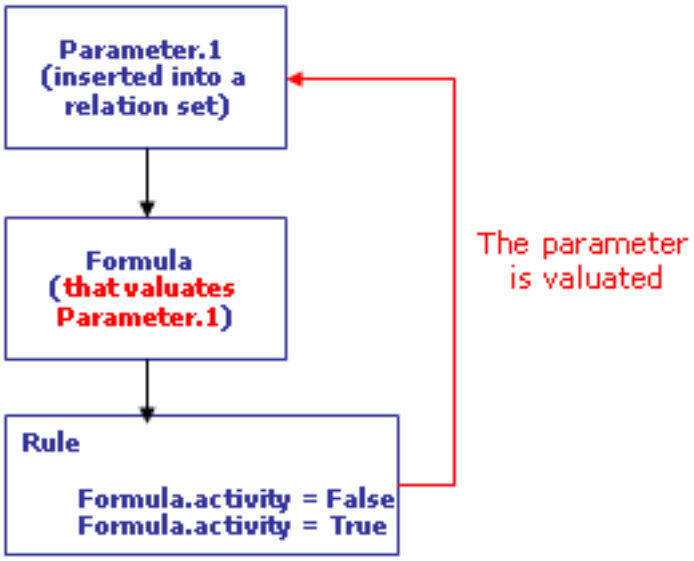
Ces relations utilisent des fonctions qui ne sont pas reconnues par le dictionnaire car la bibliothèque contenant la définition de la fonction n'est pas encore chargée. C'est pourquoi un message vous demande si vous souhaitez charger toutes les bibliothèques contenant des définitions de fonctions. Le chargement de toutes les bibliothèques peut avoir un impact sur les performances. Nous vous recommandons donc de charger uniquement les bibliothèques qui vous intéressent. Pour ce faire, sélectionnez les bibliothèques requises dans l'onglet **Langage** (**Outils->Options->Général->Knowledge->Langage**).

Règles

Fonctions et règles géométriques

Dans une règle qui utilise des fonctions nécessitant que la géométrie renvoie le type (comme les extrusions), le type ne peut pas être renvoyé lorsque cette fonction est désactivée. Pour résoudre cet incident, utilisez la commande Définir pour indiquer le type dans la règle. Pour plus d'informations, reportez-vous au fichier KwrSetType.CATPart.

Règles et cycle de mise à jour



Cette configuration est autorisée dans la mesure où une modification du paramètre d'activité n'affecte en rien la mise à jour de la formule. Toutefois :

- Dans un cas similaire, il est recommandé d'utiliser la fonction Réaction.
- Si vous souhaitez utiliser une règle, vous ne devez pas désactiver puis réactiver le paramètre d'activité.
- Lorsque vous utilisez la fonction UDF, assurez-vous d'avoir inséré l'ensemble de relations lors de la définition du gabarit.

Paramètres

- Vous pouvez ajouter des propriétés à un document .CATPart ou .CATProduct en utilisant la commande Propriétés du menu contextuel. Il suffit de cliquer sur le bouton Autres propriétés dans l'onglet Produit, puis de cliquer sur Créer paramètre de type. La boîte de dialogue qui apparaît ressemble à la boîte de dialogue *f(x)*. Pour plus d'informations, reportez-vous au document *Product Structure - Guide de l'utilisateur*. Les propriétés que vous définissez ainsi sont également affichées dans la liste des paramètres de la boîte de dialogue *f(x)*.

- Pour réordonner les paramètres appartenant à un ensemble de paramètres, utilisez la commande Réordonner... du menu contextuel.

- Les paramètres ajoutés à l'aide de l'explorateur de paramètres sont affichés juste en dessous des entités auxquelles ils sont affectés.

- Les utilisateurs *CATIA* n'utilisant pas les caractères latins doivent activer l'option **Outils->Options>Connaissance->Noms des paramètres->A encadrer avec le symbole '**. Sinon, les noms des paramètres doivent être remplacés par des noms en caractères latins lorsqu'ils sont utilisés dans des formules.

- Vous pouvez indiquer qu'un paramètre est constant à l'aide de l'option Propriétés du menu contextuel. Cette option permet en outre de masquer un paramètre.

- Lors de la copie d'ensembles de paramètres contenant des paramètres cachés, ceux-ci sont collés automatiquement lors du collage des ensembles de paramètres et apparaissent comme des paramètres cachés.

- Les paramètres ont 2 noms différents : le nom local et le nom global.

  - Le nom local est le nom attribué au paramètre lorsqu'il a été créé dans l'Editeur de formules ou dans l'explorateur de paramètres. Notez que ce nom est modifié si vous sélectionnez la commande Réordonner dans le menu contextuel.

  
  Ce nom peut être modifié à l'aide de l'explorateur de paramètres et de la commande **Edition->Propriétés**.

  - Le nom global (nom) est le nom attribué au paramètre par Knowledge Advisor. Il indique le chemin d'accès au paramètre et son type. Si vous sélectionnez le paramètre et que vous le réorganisez, le chemin contenu dans le nom sera modifié. Si vous cliquez deux fois sur le paramètre dans l'arbre de spécifications et que vous entrez un nom dans la boîte de dialogue Editer paramètre, le nom global est modifié. Si, après avoir renommé le paramètre dans la boîte de dialogue Editer paramètre, vous réorganisez le paramètre, le chemin n'apparaît plus.

  
  Ce nom peut être modifié dans la boîte de dialogue Editer paramètre.

- Lorsque vous renommez des paramètres à l'aide de la commande **Edition->Propriétés**, appliquez des noms différents de ceux des types de paramètres disponibles dans l'Editeur de formules. Par exemple, si vous renommez un paramètre Length en Time, le nom sera converti en Time.1.

- Suppression de paramètres utilisés dans une relation : Si vous supprimez un paramètre utilisé dans une relation, un paramètre "clone" sera créé.

- Application de la même formule à plusieurs paramètres : Si vous souhaitez appliquer la même formule à plusieurs paramètres, utilisez la fonction Equivalence et affectez une formule à cette fonction. Pour plus d'informations, voir Utilisation de la fonction Equivalence.



Formules

- Option Incrémental de l'Editeur de formules : L'option Incrémental vous permet de limiter la liste des paramètres affichés dans le dictionnaire. Sélectionnez une entité dans l'arbre de spécifications ou dans la zone géométrique.Seul le premier niveau des objets situés directement sous l'entité sélectionnée s'affiche dans le dictionnaire. Si l'option Incrémental est désactivée, tous les objets situés sous l'entité sélectionnée sont affichés.
  Le mode Incrémental est utile lorsque vous utilisez des documents volumineux et lorsque les listes de paramètres sont longues.

- Conseils à propos de l'Editeur de formules : Pour vous aider à écrire une formule, l'Editeur de formules vous propose un dictionnaire. Ce dictionnaire affiche la liste des paramètres et des fonctions que vous pouvez utiliser pour définir une formule. Selon la catégorie d'objets à référencer dans la formule, le dictionnaire est divisé en deux ou trois parties. Pour insérer une définition dans l'Editeur de formules, il vous suffit de cliquer deux fois sur l'objet dans le dictionnaire ou dans l'arborescence. Si vous cliquez deux fois sur une fonction dans le dictionnaire, sa signature est envoyée dans l'Editeur de formules. Seules les définitions sont absentes.



Tables de paramétrage

- Vous pouvez créer une table de paramétrage uniquement à partir de *paramètres non contraints*, c'est-à-dire des paramètres qui ne sont ni référencés dans une table de paramétrage active ni utilisés dans une autre *relation active*.
  Si vous laissez l'option Activité cochée pour DesignTable0 et que vous essayez de créer une autre table de paramétrage, vous devrez sélectionner les paramètres à ajouter à la seconde table de paramétrage dans une liste de paramètres limitée.Décochez l'option Activité si vous souhaitez désactiver une table de paramétrage et réutiliser ses paramètres dans une autre table de paramétrage.

- Chaque fois que vous modifiez une table de paramétrage, les relations qui font référence à cette table de paramétrage détectent cette modification et doivent être mises à jour.

- Tant qu'une table de paramétrage est active, les paramètres qui y sont déclarés sont des paramètres contraints. Par conséquent, vous n'êtes pas autorisé à les modifier.
  Si vous cliquez deux fois sur une table de paramétrage dans l'arbre de spécifications, la table de paramétrage s'affiche avec son ensemble de configurations et vous pouvez alors sélectionner une nouvelle configuration.

- Vous pouvez utiliser uniquement les paramètres qui ne sont pas déjà contraints par une autre relation, ou par une autre table de paramétrage, pour créer une table de paramétrage. Si un paramètre est déjà contraint, il n'apparaît pas dans la liste des paramètres à insérer dans la boîte de dialogue de la table de paramétrage.

- **Sélection des paramètres à insérer dans une table de paramétrage**
  Les filtres Filtre par Nom et Filtre par Type peuvent être utilisés pour limiter l'affichage d'une liste de paramètres. Si vous indiquez x dans le champ Filtre par nom dans la boîte de dialogue Choisissez les paramètres à insérer, tous les paramètres dont les noms contiennent la lettre x s'affichent (xA, xB, xC, xD, xE). Si vous sélectionnez le filtre Paramètres renommés dans la liste Filtre par Type, tous les paramètres que vous avez renommés dans la boîte de dialogue Formules s'affichent (yA, xB, xA, yC, xC, yB, yD, xD, yE, xE, TangE).
  Vous pouvez sélectionner simultanément plusieurs paramètres à insérer. Pour cela, il suffit de maintenir la touche Ctrl enfoncée tout en sélectionnant les paramètres. Si vous sélectionnez plusieurs paramètres simultanément, le groupe des paramètres sélectionnés est déplacé dans la liste Paramètres insérés dans l'ordre dans lequel ils sont affichés dans la liste d'origine.
  Lorsque la table de paramétrage est créée, la rangée de colonnes correspond à la rangée de paramètres affichés dans la liste Paramètres insérés. Si vous voulez organiser les colonnes d'une autre manière dans la table de paramétrage, vous devez insérer les paramètres les uns après les autres.

- **Accès aux fonctions liées à la table de paramétrage**
  Dans l'Editeur de formules (règle ou vérification), sélectionnez l'élément Table de paramétrage dans le dictionnaire. La liste des méthodes qui peuvent s'appliquer à une table de paramétrage s'affiche. Sélectionnez une méthode, puis cliquez sur la touche F1 pour afficher la documentation associée.



Boucles

- Dénomination générique : La dénomination générique est une technique de *CATIA* permettant de créer un libellé chaque fois qu'un élément est sélectionné en mode interactif. Ce libellé est une description codée de l'élément sélectionné. Lorsque vous spécifiez un congé à appliquer à une face, vous devez sélectionner la définition de cette face en mode interactif. La face à utiliser doit évidemment avoir été créée précédemment. C'est la raison pour laquelle les scripts nécessitant la définition d'une face, d'un point ou d'une arête ne peuvent être créés en une seule fois. Vous n'avez pas à vous soucier de la dénomination générique, dans la mesure où cette dernière est automatiquement capturée à partir de la zone géométrique. Par contre, vous êtes responsable de l'ordre selon lequel les instructions doivent être écrites et exécutées dans le script.

- Message "La *propriété* n'existe pas...": Vérifiez dans le navigateur que le nom de l'attribut est correct. Pour les attributs de type liste (congés et chanfreins), vérifiez les index. Ces derniers doivent être consécutifs (de 1 à n sans interruption).

- Spécification d'un chemin d'accès au fichier (deux méthodes)

Syntaxe

Il existe deux manières d'indiquer le chemin du fichier :

- **import "File path";** : Indique le chemin d'accès au fichier à importer : `import "E:\users\kwecx\Models\PartImport.CATPart" ;`. Vous devrez entrer le chemin entre guillemets et terminer l'instruction d'importation par un point virgule (;).
- **import "File Name";** : Indique uniquement le nom du fichier à importer si ce fichier est situé dans le même répertoire que le document contenant la boucle : `import "PartImport.CATPart";`. Remarques :
  - Vous devrez entrer le nom du document entre guillemets et terminer l'instruction d'importation par un point virgule (;).
  - Le fichier à importer doit résider dans le même répertoire que celui du document contenant la boucle.
  - Le document contenant la boucle doit être enregistré.

Pour indiquer un fichier à importer, vous pouvez utiliser la commande Chemin d'accès au fichier à partir du menu contextuel. Lorsque vous sélectionnez cette commande, une fenêtre de sélection de fichier s'affiche. Seuls les guillemets sont automatiquement inclus (le point-virgule ne l'est pas).

- Importation d'esquisses : Recommandations

Lors de la conception d'un document devant être créé par le biais d'un script, il est préférable de regrouper toutes les esquisses dans un fichier unique. Vous devez procéder de la manière suivante :

- Réduisez la taille globale des données relatives aux esquisses.
- Quelle que soit la méthode utilisée pour spécifier le fichier d'entrée, vous ne devez indiquer le chemin d'accès qu'une seule fois.
- La conception du document final devient alors plus simple. Vous obtenez une vue générale des esquisses sur lesquelles les autres entités reposent.

- Spécification de chaînes : Recommandation : Des guillemets doubles ainsi que des guillemets simples de type **apostrophe** `) peuvent être utilisés pour délimiter les chaînes. Des guillemets simples (`) doivent être utilisés pour encadrer les chaînes de caractères contenant d'autres chaînes.

Réactions

Il existe deux manières de réagir aux modifications de paramètre à l'aide de la fonction Réaction :

- Sélectionnez le paramètre et réagissez à l'événement Value Changed

Ou

- Sélectionnez l'entité (qui possède l'attribut correspondant à votre paramètre) et réagissez à l'événement Attribute Modification. Notez que cette méthode ne fonctionne pas dans tous les cas.

Exemples d'utilisation

Roulement à billes

Système de trois équations dans trois variables

Roulement à billes

Un roulement se définit au travers de paramètres tels que ses cotes principales, ses caractéristiques de charge, ses vitesses limites et sa masse. Ce sont les valeurs de ces paramètres qui déterminent la catégorie à laquelle le roulement appartient. Dans un catalogue, un roulement est identifié par une désignation. Les différents types de roulement sont décrits dans des tableaux qui indiquent les valeurs des paramètres et les désignations.

Nous avons choisi l'exemple du roulement parce que les tableaux de spécification que vous trouvez dans les catalogues des fabricants et revendeurs illustrent assez bien les principes des tables de paramétrage. En outre, le roulement lui-même est un bon exemple pour expliquer comment les composants d'une pièce mécanique peuvent être assortis de contraintes par le biais de relations.

Dans l'exemple ci-dessous, le document de départ s'inspire d'un roulement à billes à rainures profondes. Ce document contient déjà plusieurs relations de règles de connaissance ; il existe des relations supplémentaires pour contrôler la conception du document.

Préambule
Pas à pas

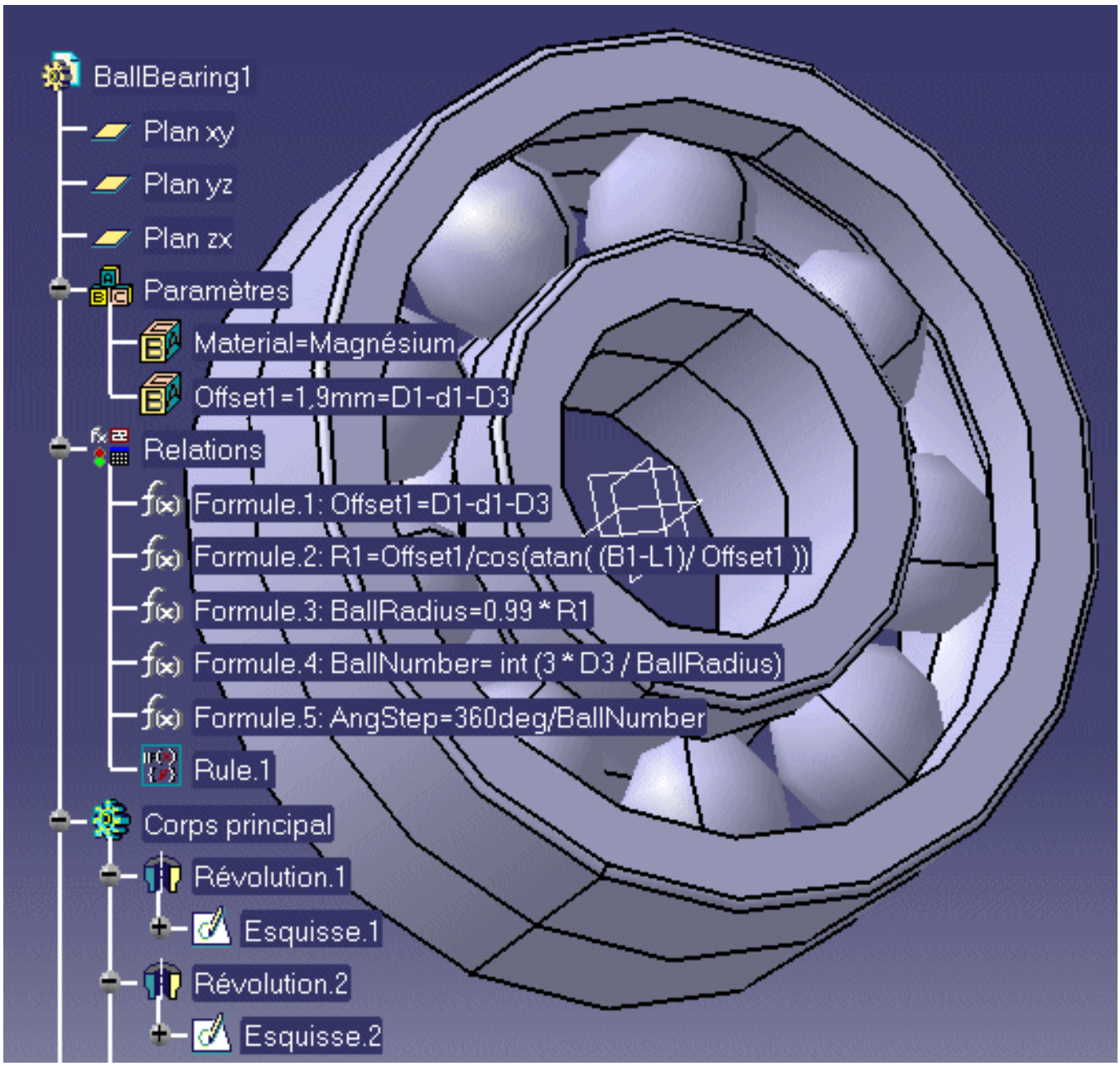
Préambule

Cette section décrit les données nécessaires au bon déroulement du scénario. Elles sont toutes fournies avec le programme Knowledge Advisor mais peuvent être recréées à partir des informations données ci-dessous.

Pour plus d'informations sur la définition des paramètres du catalogue de matériaux, reportez-vous au document *Infrastructure - Guide de l'utilisateur* (vous devez utiliser la commande **Outils->Options...->Infrastructure->Material Library** à partir de la barre de menus standard).

Document de départ

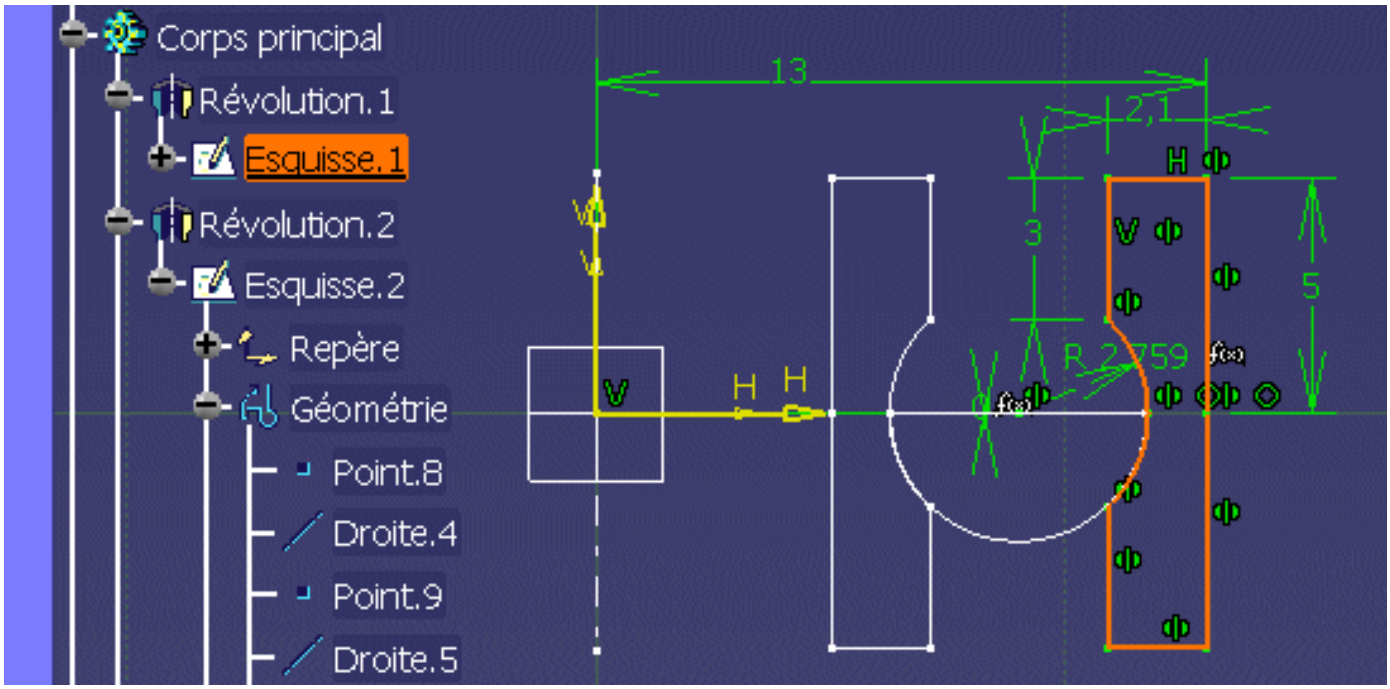
Le document de départ est KwrBallBearing1.CATPart.



Les anneaux du roulement sont des axes coaxiaux créés à partir des composants Esquisse.1 et Esquisse.2. Les billes sont des axes créés à partir de l'entité Esquisse.3.

Anneau externe

L'anneau externe est un axe créé par rotation de l'entité Esquisse.1 mise en évidence dans la figure ci-dessous, autour d'un axe coaxial à V. Notez que vous devez créer cet axe en tant qu'*élément de construction*, faute de quoi *CATIA* ne vous permettra pas de créer l'axe. Les parties inférieure et supérieure de l'esquisse sont symétriques par rapport à l'axe H.

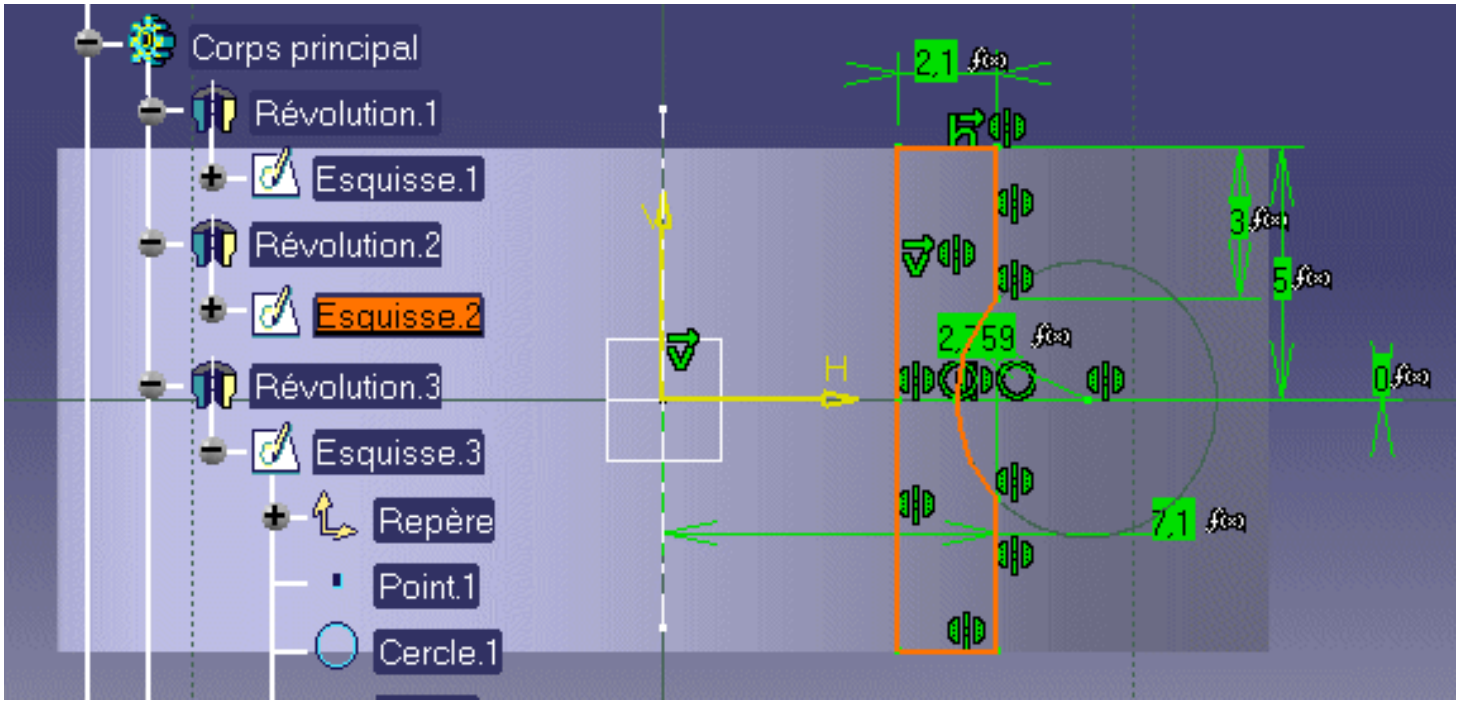


Les contraintes de cette esquisse sont les suivantes :

d1	2,1 mm	*largeur de l'anneau*
L1	3 mm	*mi-hauteur de la surface interne non creusée*
B1	5 mm	*mi-hauteur de la surface extérieure*
R1	2,759 mm	*rayon des rainures*
b1	0 mm	*ordonnée du centre des rainures*
D1	13 mm	*diamètre externe*

Anneau interne

L'anneau interne est un axe créé par rotation de l'entité Esquisse.2 mise en évidence dans la figure ci-dessous, autour d'un axe coaxial à V.

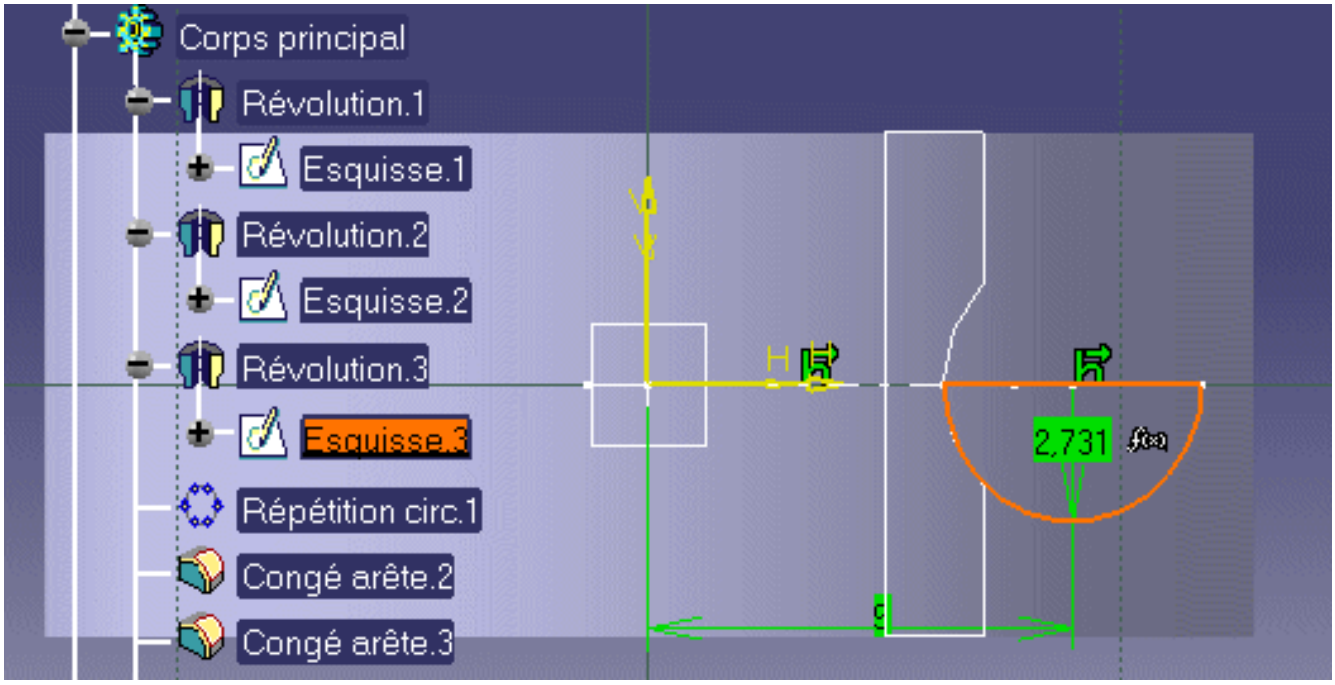


Les contraintes de cette esquisse sont les suivantes :

d2	2,1 mm	*largeur de l'anneau*
L2	3 mm	*mi-hauteur de la surface interne non creusée*
B2	5 mm	*mi-hauteur de la surface extérieure*
R2	2,759 mm	*rayon des rainures*
b2	0 mm	*ordonnée du centre des rainures*
D2	7,1 mm	*diamètre interne*

Billes

Une bille est un axe créé par rotation d'un demi-cercle (esquisse.3) autour de l'axe H. Vous devez fermer le cercle avant d'effectuer la rotation.



Les paramètres du modèle circulaire créé pour générer un ensemble de billes sont assortis des formules de contrainte suivantes :

- BallNumber = int(3* D3 / BallRadius)
- AngStep = 3.6deg / BallNumber

D3 étant l'abscisse du centre de la bille.

Fichier d'importation

Dans l'exemple, vous devez importer le fichier texte fourni comme modèle KwrBallBearingImport.txt.

```
Temperature     100Kdeg         Maximum temperature allowed
Pressure        190N_m2         Maximum pressure allowed
LubricantVolume 0mm3                    L1*D3*B1*0.005    required lubricant volume
```

Si vous modifiez ce fichier, prêtez attention au format des colonnes et utilisez la touche de tabulation pour passer d'une colonne à une autre.

Tableau Excel gérant la conception du roulement

Vous devez télécharger le tableau Excel KwrBearingDesignTable.xls dans votre environnement.

Macro CATScript

La macro KwrBearing.CATScript crée uniquement une extrusion circulaire. Vous pouvez vous-même enregistrer cette macro dans l'atelier Part Design ou utiliser celle fournie avec les modèles KnowledgeAdvisor.

Lorsque vous créez la règle **Règle2** dans votre propre environnement, remplacez le chemin d'accès donné comme argument de la fonction LaunchMacroFromFile par celui du répertoire contenant la macro.

Procédure pas à pas

Contrôle de la conception d'un roulement à l'aide d'une table de paramétrage

Une table de paramétrage est créée sur la base d'un fichier existant. Les données contenues dans ce fichier sont semblables à celles qui caractérisent un roulement dans un catalogue. La table de paramétrage créée définit un certain nombre de configurations. L'application d'une nouvelle configuration entraîne la modification du roulement.

1. Ouvrez le document KwrBallBearing1.CATPart.
2. Cliquez sur l'icône Table de paramétrage  dans la barre d'outils standard.
3. Cochez l'option **Créer une table de paramétrage à partir d'un fichier existant**. Cliquez sur **OK**.
4. Sélectionnez le fichier KwrBearingDesignTable.xls et faites une association automatique entre les colonnes de la table de paramétrage et les paramètres du document (c'est-à-dire cliquez sur **OUI** dans la boîte de dialogue "Associations automatiques ?").
5. Dans la boîte de dialogue **Table de paramétrage**, sélectionnez la configuration 3 (Droite 3) et cliquez sur **Appliquer**.
   Le roulement à billes a changé. Il est à présent en bronze et contient 21 billes. Vous pouvez remarquer la différence lorsque vous regardez la zone géométrique. La largeur du roulement a également été modifiée. Cliquez sur **OK** pour quitter la boîte de dialogue Table de paramétrage.
6. Gardez le document ouvert et passez à la tâche suivante.

Création d'une vérification

CATIA crée une vérification combinée à l'aide de la syntaxe =>. Cette vérification affiche un message chaque fois qu'elle échoue.

1. Ouvrez l'atelier Knowledge Advisor.
2. Cliquez sur l'icône  , puis sur **OK** dans la boîte de dialogue **Editeur de vérifications**. L'éditeur de vérifications s'affiche.
3. Dans l'**éditeur de vérifications**, sélectionnez le type Warning et entrez la chaîne "Le nombre de billes est trop faible" dans la zone de message.

   Entrez ensuite la relation

   D3 >= 6mm => BallNumber > 6

   dans la zone d'édition.
4. Cliquez sur OK pour confirmer la création de la vérification et quitter l'éditeur. A ce stade, aucun message ne s'affiche. La vérification ainsi qu'une icône verte sont ajoutées à l'arbre de spécifications. Pour la configuration 3 de la table de paramétrage, le statut des relations de la vérification est le suivant :

   OK => OK 
5. Dans l'arbre de spécifications, cliquez deux fois sur la table de paramétrage et sélectionnez la configuration 1. Cliquez sur **OK**. Le message "Le nombre de billes est trop faible" s'affiche. Pour la configuration 1 de la table de paramétrage, le statut des relations de la vérification est le suivant :

   OK => KO 

Gardez le document ouvert et passez à la tâche suivante.

Création d'un paramètre à plusieurs valeurs

Un paramètre à plusieurs valeurs est créé. En fonction de la valeur de ce paramètre, une règle créée lors de la tâche suivante affiche un message ou exécute une macro.

1. Cliquez sur l'icône  .
2. Dans la boîte de dialogue **Formules**, sélectionnez Chaîne dans la liste **Créer paramètre de type**. Dans la liste Avec, sélectionnez **Valeurs multiples**, puis cliquez sur '**Créer paramètre de type**'.
3. Dans la boîte de dialogue Liste de valeurs de chaîne, entrez une à une les valeurs step1, step2 et step3. Cliquez sur OK.
4. Dans la boîte de dialogue **Editer le nom ou la valeur du paramètre sélectionné**, remplacez la chaîne String.1 par Etat, puis cliquez sur **OK**. Le paramètre Etat est ajouté à l'arbre de spécifications.

Création d'une règle

Cette tâche crée une règle qui affiche un message vous demandant d'importer un fichier ou qui exécute une macro.

1. Dans l'arbre de spécifications, cliquez deux fois sur l'entité de la table de paramétrage et sélectionnez la configuration 3 dans la table qui s'affiche. Le roulement contient à nouveau 21 billes.
2. Ouvrez l'atelier Knowledge Advisor.
3. Cliquez sur l'icône  .
4. Entrez la chaîne **Règle.2** dans le champ Nom de la première boîte de dialogue. Cliquez sur **OK**.
5. Insérez le code ci-dessous dans la zone d'édition des règles avec un copier-coller (modifiez le chemin de la macro) :

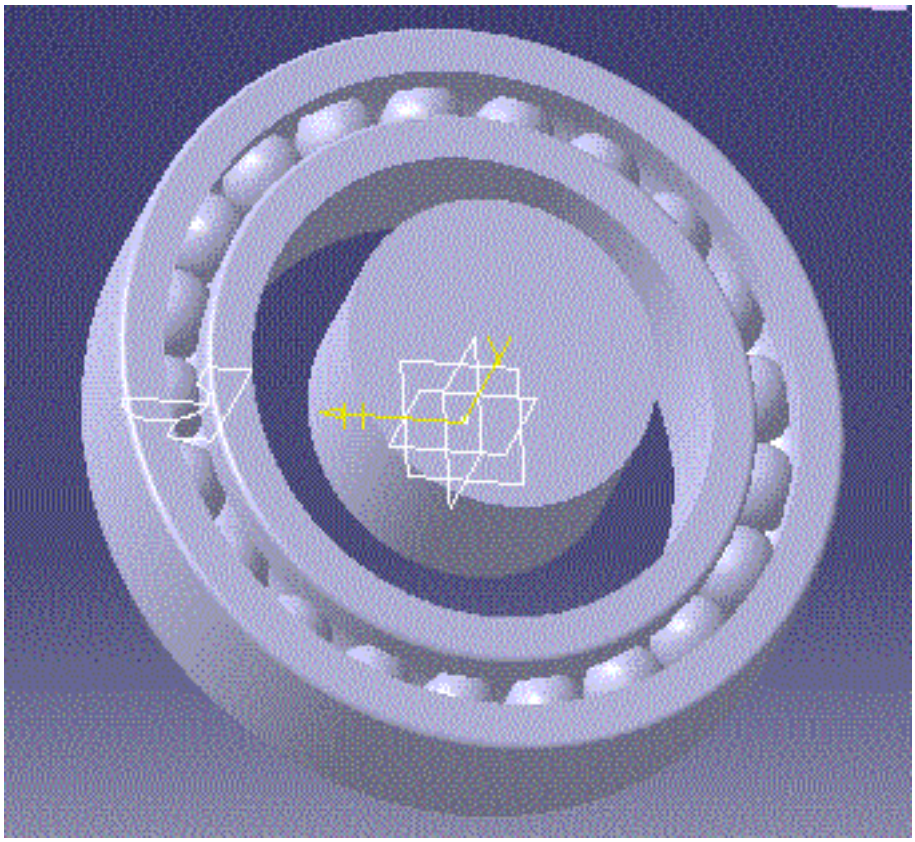
```
if Etat == "step2"
Message("Importez le fichier texte KwrBallBearingImport")
else if Etat == "step3"
LaunchMacroFromFile("e:/tmp/KwrBearing.CATScript")
```

6. Cliquez sur **OK** pour ajouter la règle au document et exécutez-la.

7. Cliquez sur l'icône  . Dans la boîte de dialogue Formules, sélectionnez le paramètre Etat et remplacez la valeur step1 par step2. Cliquez sur **OK**. Un message vous demande d'importer le fichier texte KwrBallBearingImport.

8. Cliquez sur **Importer** et sélectionnez le fichier KwrBallBearingImport.txt. Trois paramètres viennent s'ajouter au document. Cliquez sur OK dans la boîte de dialogue contenant les paramètres et les formules à importer.

9. Sélectionnez le paramètre Etat et remplacez la valeur step2 par step3. Cliquez sur **OK**. Le fichier KwrBearing.CATScript s'exécute et crée une extrusion circulaire.



Système de trois équations dans trois variables

Lors de la conception d'un produit, il se peut que vous ayez un système d'équations à résoudre. Quelles que soient ces équations (linéaires ou non), *CATIA* fournit des méthodes de résolution. Ces méthodes sont l'***Algorithme de recuit simulé*** et la fonction ***Ensemble d'équations***.

Pouvez-vous utiliser n'importe laquelle de ces méthodes ?
Si l'ensemble d'équations est purement mathématique, la réponse est oui. Dans le cas contraire, la réponse est non. La fonction Ensemble d'équations ne peut pas résoudre les systèmes utilisant des fonctions *CATIA* telles que les mesures.
Pour résoudre un système d'équations utilisant des mesures, vous devez utiliser l'algorithme de recuit simulé.
L'algorithme de recuit simulé est fourni avec le programme Knowledge Optimizer. L'ensemble d'équations et les variables doivent être définis respectivement comme contraintes et paramètres libres. Cette méthode de résolution est très efficace quoique relativement longue et vous pouvez l'utiliser pour résoudre un bon nombre de cas. La particularité de cet algorithme est qu'il demande d'ajuster la précision et les autres paramètres. L'exemple développé ci-après fonctionne parfaitement avec les deux méthodes. Pour illustrer un système qui ne peut pas être résolu par les deux méthodes, vous pouvez dessiner un cube et créer deux paramètres utilisateur : CubeSurface (de type Aire) et CubeVolume (de type Volume). Pour calculer CubeSurface et CubeVolume, vous pouvez écrire au choix :

```
CubeSurface = smartWetarea ( PartBody\Pad.1 )
CubeVolume = smartVolume ( PartBody\Pad.1 )
```

ou

```
CubeVolume = smartVolume ( PartBody\Pad.1 )
```

Résolution d'un système d'équations avec un recuit simulé

1. Ouvrons un nouveau document Part.
2. Créez six paramètres de type réel avec les fonctions f(x). Donnez un nom à ces paramètres :

   x1, y1, z1 et x2, y2, z2.

3. Ouvrez Knowledge Optimizer et cliquez sur l'icône  .
4. Sous l'onglet Contraintes, indiquez les trois contraintes (entrez-les une à une) :

   x1 + y1 - z1 == 0

   x1*y1 - z1 == 0

   sin(x1*1rad)2 - y1 - 1 == 0**

Indiquez une précision de 0,01 pour les trois contraintes.

Le cas échéant, reportez-vous au document *Knowledge Optimizer - Guide de l'utilisateur*.

5. Sous l'onglet Problème, définissez x1, y1, z1 comme paramètres libres et affectez la valeur 1 comme valeur de pas pour les trois paramètres.

6. Exécutez le processus d'optimisation en mode Recuit simulé. Vous pouvez utiliser le critère d'arrêt par défaut. Une fois le processus exécuté, les valeurs des paramètres x1, y1 et z1 se rapprochent des suivantes :

```
x1 = 0,454
y1 = -0,807
z1 = -0,363
```

Gardez le document ouvert et passez à la tâche suivante.

Résolution d'un système d'équations avec la fonction "Ensemble d'équations"

1. Ouvrez l'atelier Knowledge Advisor, puis cliquez sur l'icône  .

2. Dans l'éditeur "Ensemble d'équations" entrez l'ensemble d'équations suivant :

```
x2 + y2 == z2 ;

x2*y2 == z2;

sin(x2*1rad)**2 == y2 +1
```

Définissez x2, y2 et z2 comme Paramètres inconnus à l'aide du bouton de la flèche Analyse ().

3. Cliquez sur OK. Le système d'équations est résolu. Les valeurs ci-après apparaissent dans l'arbre de spécifications.

```
x2 = 0,448043478

y2 = -0,812335288

z2 = -0,364229828
```

Interopérabilité de Knowledge Advisor

Dans cette tâche, vous apprendrez à créer des relations Knowledge Advisor à l'aide de modules technologiques et à les stocker dans ENOVIA LCA.

Pour plus d'informations sur les modules technologiques, voir le document *CATIA Infrastructure - Guide de l'utilisateur*.

Utilisation optimale de CATIA PLM pour Knowledge Advisor
Sauvegarde d'une structure de produit contenant une règle dans ENOVIA LCA

Utilisation optimale de CATIA PLM pour Knowledge Advisor

Lorsque vous utilisez ENOVIA LCA, le mode de sauvegarde sécurisée vous permet de ne créer dans *CATIA* que des données pouvant être correctement enregistrées dans ENOVIA LCA.

ENOVIA LCA propose deux modes de stockage différents : le mode Espace de travail (document conservé - publications affichées) et le mode Vue éclatée (document non conservé).

Commandes Knowledge Advisor dans Enovia LCA

Vous trouverez ci-dessous la liste des commandes Knowledge Advisor ainsi que leur état d'accessibilité dans ENOVIA LCA.

Les composants pris en charge par Knowledge Advisor sont les suivants :

Commandes	Accessibilité dans Enovia V5 (mode Vue éclatée)	Commentaires
Règle	Disponible	Aucun
Vérification	Disponible	Aucun
Réactions	Disponible	Aucun
Liste	Disponible	Aucun
Boucle	Disponible	Aucun
Ajout d'ensemble de paramètres	Disponible	Aucun
Ajout d'ensemble de relations	Disponible	Aucun
Explorateur de paramètres	Disponible	Aucun
Commentaire & URLs	Disponible	Aucun
Macros avec arguments	Disponible	Aucun
Actions	Disponible	Aucun
Mise à jour de la mesure	Disponible	Aucun
Ensemble d'équations	Disponible	Aucun

Sauvegarde d'une structure de produit contenant une règle dans ENOVIA LCA

Dans cette tâche, vous apprendrez à sauvegarder un produit contenant une règle dans ENOVIA LCA. Dans ce scénario, vous allez :

- créer un produit dans ENOVIA LCA et l'envoyer dans CATIA ;
- ajouter deux composants au produit ainsi qu'une règle qui pilote les deux composants ajoutés, puis stocker le produit dans ENOVIA LCA.

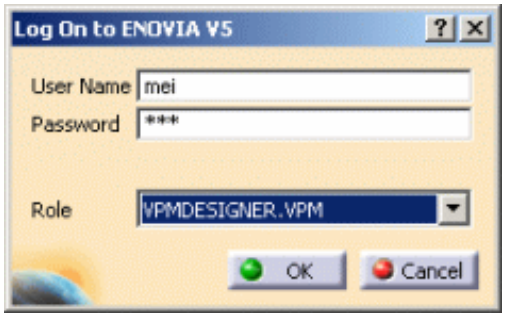
- Assurez-vous d'avoir sélectionné **ENOVIAV5** dans le menu **Affichage -> Barres d'outils**.

1. Créez un produit dans ENOVIA LCA. (dans ce scénario, le produit s'appelle Advisor).

   Vous pouvez fermer ENOVIA après avoir créé le produit.

Envoi du produit à CATIA

2. Pour envoyer le produit à l'application CATIA :

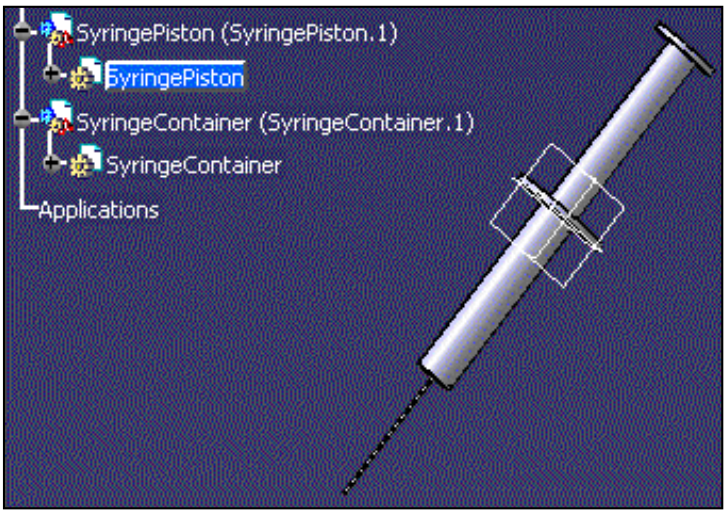


   - Dans CATIA, cliquez sur l'icône **Connexion** () dans la barre d'outils **ENOVIA V5**. La boîte de dialogue de connexion s'affiche.
   - Entrez votre nom d'utilisateur et votre mot de passe.
   - Sélectionnez votre rôle dans la liste déroulante contenant les rôles et cliquez sur **OK**.
   - Cliquez sur l'icône **Rechercher les données ENOVIA** (). La boîte de dialogue Recherche VPM s'affiche.
   - Dans la liste déroulante Objets, sélectionnez l'option **Produit**.
   - Entrez l'ID du propriétaire.
   - Lorsque vous avez terminé, cliquez sur **OK**. L'ID produit s'affiche dans la fenêtre Résultat.
   - Cliquez deux fois sur le produit. La fenêtre Navigateur VPM s'affiche.
   - Cliquez avec le bouton droit de la souris sur le produit racine (Advisor) et sélectionnez la commande **Ouvrir**. La boîte de dialogue Modes d'ouverture s'affiche. Conservez les paramètres par défaut et cliquez sur **OK**. La fenêtre CATIA s'affiche.

Ajouts de composants au produit

3. Cliquez sur le produit racine et sélectionnez la commande **Insertion -> Composant existant...**. Sélectionnez le document KwrSyringeContainer.CATPart. La boîte de seringues est insérée dans le produit.

4. Cliquez sur le produit racine et sélectionnez la commande **Insertion -> Composant existant...**. Sélectionnez le document KwrSyringePiston.CATPart. La boîte à seringues est insérée dans le produit.



Ajout d'une règle au produit

5. Dans le menu **Démarrer->Knowledgeware**, ouvrez l'atelier **Knowledge Advisor** et cliquez sur l'icône **Règle** ( ) pour créer une règle.
6. Entrez le corps de la règle ci-après à l'aide des paramètres disponibles dans la colonne Membres de dans l'Editeur de règles, puis cliquez sur **OK** lorsque vous avez terminé.

```
SyringeContainer\PartBody\Sketch.2\Radius.2\Radius =
SyringePiston\PartBody\Sketch.2\Radius.2\Radius +5mm
SyringeContainer\PartBody\Sketch.1\Radius.1\Radius =
SyringePiston\PartBody\Sketch.1\Radius.1\Radius
```

Les utilisateurs japonais doivent entrer la règle suivante :

```
`SyringeContainer¥ﾊﾟｰﾂ ﾎﾞﾃﾞｨｰ¥Sketch.2¥半径.2¥半径` = `SyringePiston¥ﾊﾟｰﾂ ﾎﾞﾃﾞｨｰ¥Sketch.2¥半径.2¥半径` +5mm
`SyringeContainer¥ﾊﾟｰﾂ ﾎﾞﾃﾞｨｰ¥Sketch.1¥半径.1¥半径` = `SyringePiston¥ﾊﾟｰﾂ ﾎﾞﾃﾞｨｰ¥Sketch.1¥半径.1¥半径`
```

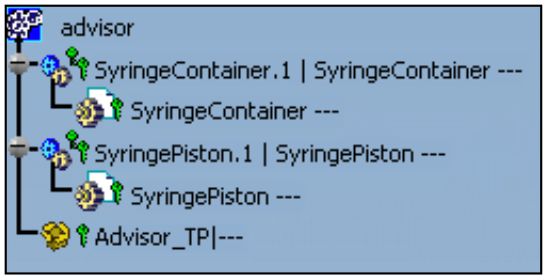
La règle est appliquée au document.

7. Dans le menu **Démarrer->Conception mécanique**, accédez à l'atelier **Assembly Design** et cliquez sur l'icône **Mise à jour complète** ( ) pour mettre à jour le document.
8. Cliquez avec le bouton droit de la souris sur le module technologique et sélectionnez la commande **Propriétés**. Renommez ce module Advisor_TP dans le scénario.

Enregistrement des données dans ENOVIA LCA

9. Cliquez sur l'icône **Enregistrer des données dans ENOVIA V5** ( ). La boîte de dialogue Sauvegarde dans ENOVIA V5 s'affiche. Cliquez sur **OK**. Votre document est sauvegardé dans ENOVIA. Fermez le produit dans CATIA.
10. Envoyez à nouveau votre produit dans ENOVIA.
    - Fermez toutes les fenêtres dans CATIA.

- Cliquez sur l'icône **Rechercher les données ENOVIA** (). La boîte de dialogue Recherche VPM s'affiche.
- Dans la liste déroulante Objets, sélectionnez l'option **Produit**.
- Entrez l'ID du propriétaire.
- Lorsque vous avez terminé, cliquez sur **OK**. L'ID produit s'affiche dans la fenêtre Résultat.
- Cliquez deux fois sur le produit. La fenêtre Navigateur VPM s'affiche. Ouvrez l'assemblage.

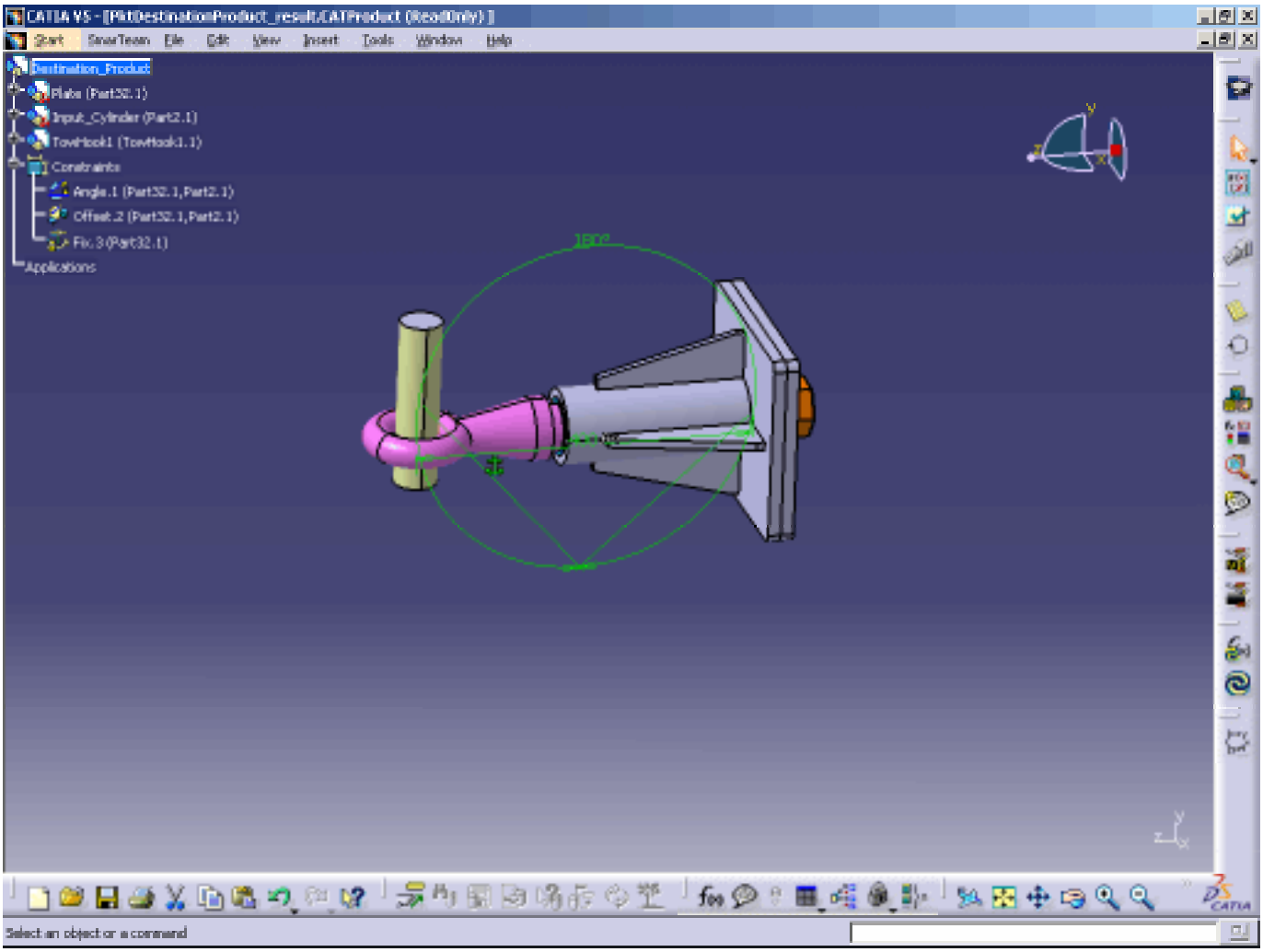




Description de l'atelier

Cette section contient la description des icônes et des menus propres à l'atelier Knowledge Advisor.

L'atelier Knowledge Advisor est illustré ci-dessous. Cliquez sur les zones sensibles (barres d'outils) pour accéder à la documentation associée.



Barre de menus de Knowledge Advisor
Barre d'outils de Knowledge Advisor
Barre d'outils Objets réactifs
Barre d'outils Objets de contrôle
Barre d'outils Organisation de la connaissance
Barre d'outils Actions
Barre d'outils Outils
Barre d'outils Objets ensemble d'équations

Barre de menus de Knowledge Advisor

Les différents menus et commandes de menus spécifiques à Knowledge Advisor sont décrits ci-après.

Démarrer Fichier Edition Affichage Insertion Outils Fenêtre ?

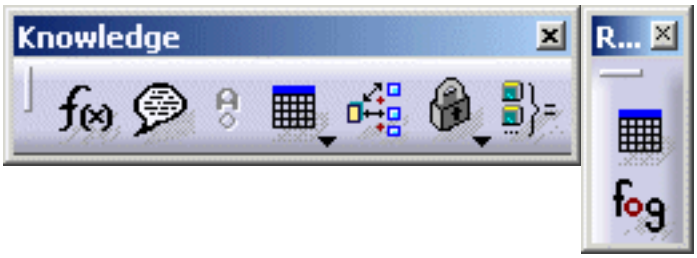
Outils



Pour	Voir
Formule...	Création d'une formule

Barre d'outils de Knowledge Advisor

La barre d'outils de Knowledge Advisor contient les outils suivants :





Voir Création d'un paramètre et Création d'une formule.



Voir Association d'adresses URL et de commentaires à des paramètres ou à des relations.



Voir Création d'une table de paramétrage à partir de valeurs courantes et Création d'une table de paramétrage à partir d'un fichier existant.



Voir Création d'une loi.



Voir Utilisation de l'inspecteur de gestion de connaissance.



Voir Utilisation de la fonction Equivalence.



Voir Verrouillage et déverrouillage d'un paramètre.

Barre d'outils Objets réactifs

La barre d'outils Objets réactifs contient les outils suivants :





Voir Création d'une règle.



Voir Création d'une vérification.



Voir Création d'une réaction : événement DragAndDrop Event.

Barre d'outils Organisation de la connaissance

La barre d'outils Organisation de la connaissance contient les outils suivants :





Voir Création d'ensembles de paramètres.



Voir Création d'ensembles de relations.



Voir Ajout d'un paramètre à une entité.



Voir Ajout d'un paramètre à une arête.



Voir Association d'adresses URL et de commentaires à des paramètres ou à des relations.

Barre d'outils Objets de contrôle

La barre d'outils Objets de contrôle contient les outils suivants :



Ces icônes sont sommairement décrites ci-après.



Voir Création d'une liste.



Voir Création d'une boucle.

Barre d'outils Actions

La barre d'outils Actions contient les outils suivants :





Voir Lancement d'une macro VB avec des arguments.



Voir Utilisation de la fonction Action de Knowledge Advisor.

Barre d'outils Outils

La barre d'outils Outils contient les outils suivants :





Voir Mise à jour de relations utilisant des mesures.



L'icône Mise à jour permet de mettre à jour un document sans quitter l'atelier Knowledge Advisor.

Barre d'outils Objets ensemble d'équations

La barre d'outils Objets ensemble d'équations contient les outils suivants :





Voir Résolution d'un ensemble d'équations.

Personnalisation

Avant de démarrer votre première session de travail, vous pouvez personnaliser votre manière de travailler en fonction de vos habitudes.
Ce type de personnalisation est stocké de manière permanente : les paramètres seront conservés si vous quittez votre session.

1. Sélectionnez la commande **Outils -> Options**. La boîte de dialogue **Options** s'affiche.
2. Cliquez sur la catégorie **Général** dans l'arborescence de gauche.
3. Cliquez sur l'onglet **Paramètres et mesure**. Les onglets suivants s'affichent.



Cet onglet vous permet de définir ce qui suit :

- les paramètres Knowledge
- les bibliothèques à charger
- les paramètres de création de rapport

4. Deux autres onglets, situés dans la catégorie Infrastructure dans l'atelier Part Infrastructure, sont également communs aux applications Knowledgeware.



- Général
- Affichage

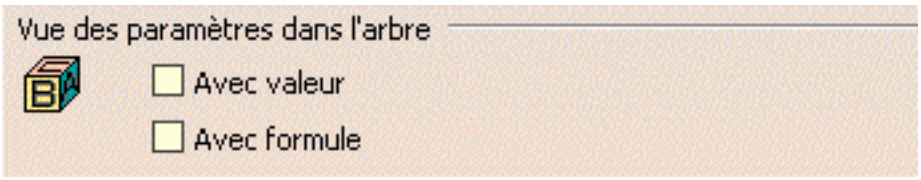
5. Vous pouvez modifier ces options en fonction de vos besoins.
6. Lorsque vous avez terminé, cliquez sur **OK** pour valider la sélection.

Connaissance

Cette section traite des catégories d'options suivantes :

- Vue des paramètres dans l'arbre
- Noms des paramètres
- Mise à jour des relations en contexte de pièce
- Tables de Paramétrage

Vue des paramètres dans l'arbre



Vous pouvez afficher deux types d'élément dans l'arbre de spécifications.

Avec valeur

Permet d'afficher les valeurs des paramètres dans l'arbre de spécifications.

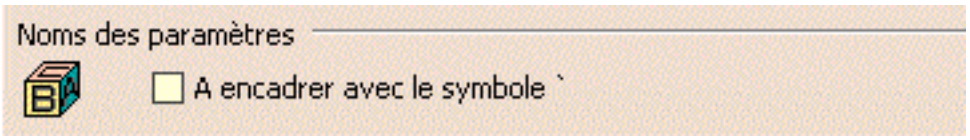
Par défaut, cette option est désactivée.

Avec formule

Permet d'afficher les formules imposant des contraintes aux paramètres dans l'arbre de spécifications.

Par défaut, cette option est désactivée.

Noms des paramètres

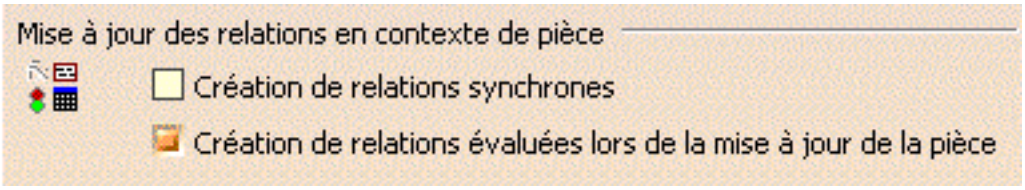


Cette option doit être activée si vous utilisez des caractères autres que les caractères latins.Si cette option n'est pas activée, les noms des paramètres doivent être remplacés par des noms en

caractères latins lorsque vous les utiliserez dans des formules.

Par défaut, cette option est désactivée.

Mise à jour des relations dans les pièces



Avant la version 5 R 12, les relations Knowledge (formules, règles, vérifications, tables de paramétrage et ensembles d'équations) s'exécutaient dès modification des données associées. L'utilisateur peut désormais choisir lors de la création de la relation si elle est synchrone (calcul effectué dès modification des paramètres) ou asynchrone (calcul effectué lors de la mise à jour de la pièce). Chaque relation peut donc être synchrone ou asynchrone.

Les deux options suivantes permettent de créer des relations synchrones ou asynchrones.

Création de relations synchrones

Permet de créer des relations synchrones, c'est-à-dire des relations mises à jour immédiatement lors de la modification de l'un des paramètres/l'une des entrées. Seules les relations basées sur les paramètres peuvent être synchrones.

Par défaut, cette option est désactivée.

Création de relations évaluées lors de la mise a jour de la pièce

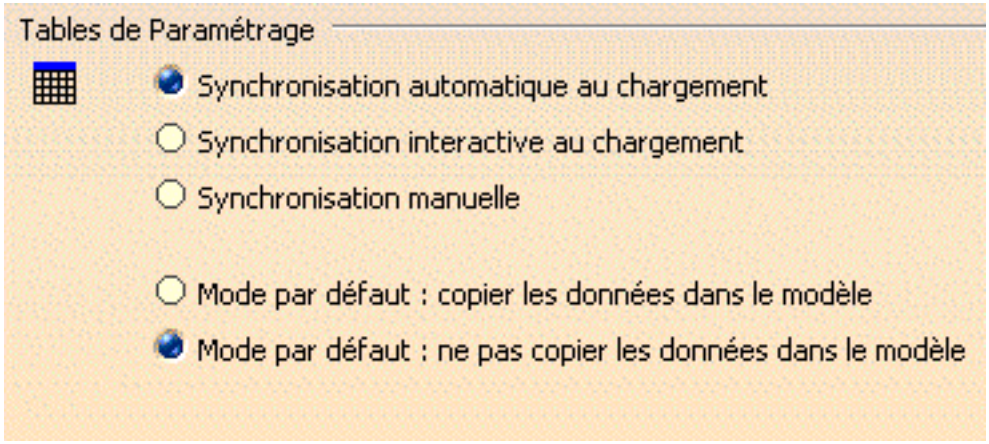
Permet d'associer l'évaluation des relations asynchrones avec la mise à jour globale. Les relations peuvent être asynchrones pour deux raisons :

- L'utilisateur souhaite qu'elles le soient.
- La relation contient des mesures.
- Relations basées sur les paramètres : Ces relations peuvent être synchrones ou asynchrones.
- Relations basées sur la géométrie : Ces relations peuvent uniquement être asynchrones.
- Relations basées sur les paramètres et sur la géométrie : L'utilisateur doit décider si la mise à jour de la partie de la relation contenant des paramètres doit être synchrone ou non. En ce qui concerne l'autre partie des relations, la mise à jour se produit lorsque la mise à jour globale est lancée.

L'utilisateur peut en outre décider si des relations existantes sont synchrones ou asynchrones. Pour plus d'informations, voir Contrôle de la mise à jour des relations dans Infrastructure - Guide de l'utilisateur.

Par défaut, cette option est activée.

Tables de Paramétrage



Vous pouvez paramétrer deux types d'élément.

Synchronisation automatique au chargement

Lorsque vous chargez un modèle contenant des tables de paramétrage d'utilisateur, si les fichiers des tables ont été modifiés et que des données de fichiers externes sont présentes dans le modèle, la table de paramétrage est automatiquement synchronisée si ce bouton d'option est activé.

Par défaut, cette option est activée.

Synchronisation interactive au chargement

Lorsque vous chargez un modèle contenant des tables de paramétrage d'utilisateur dont le fichier source externe a été supprimé, cette option permet à l'utilisateur de sélectionner un nouveau fichier source ou de sauvegarder les données des tables de paramétrage dans un nouveau fichier.

Par défaut, cette option est désactivée.

Synchronisation manuelle

Lorsque vous chargez un modèle contenant des tables de paramétrage d'utilisateur, si les fichiers des tables ont été modifiés et que des données de fichiers externes sont présentes dans le modèle, la table de paramétrage est synchronisée si ce bouton d'option est activé. Pour synchroniser les deux fichiers, cliquez à l'aide du bouton droit de la souris sur la table de paramétrage dans l'arbre de spécifications, puis sélectionnez la commande **DesignTable object->Synchronize** ou la commande **Edition->Liaisons**.

Par défaut, cette option est désactivée.

Mode par défaut : copier les données dans le modèle

Si cette option est cochée, les données contenues dans le fichier source externe sont copiées dans le modèle.

Par défaut, cette option est désactivée.

Mode par défaut : ne pas copier les données dans le modèle

Si cette option est cochée, les données contenues dans le fichier source externe ne sont pas copiées dans le modèle.

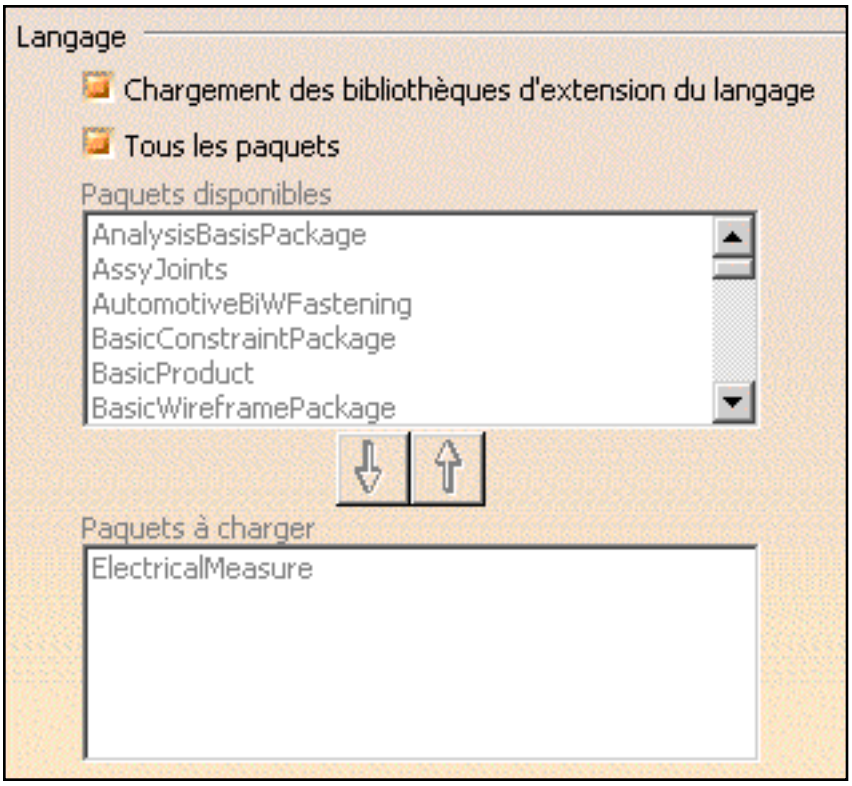
Par défaut, cette option est activée.

Langage

Cette section traite des catégories d'options suivantes :

- Langage
- Répertoire de référence pour les types

Langage



Utilisez cette zone lorsque vous utilisez des *mesures* dans des relations ou des *fonctions utilisateur*. Les mesures sont des fonctions spécifiques utilisées dans des formules et règles.

Le document *Knowledge Advisor - Guide de l'utilisateur* contient des sections relatives à l'utilisation des mesures. Pour savoir comment créer et utiliser les fonctions utilisateur, voir la documentation *CATIA* Application Architecture.

Chargement des bibliothèques d'extension du langage

Lorsqu'elle est activée, cette option permet de sélectionner les paquets à charger sous Paquets (si l'utilisateur souhaite charger un nombre limité de paquets).

Cette fonction s'avère particulièrement utile à l'administrateur pour limiter le nombre de paquets qu'utilise l'utilisateur.Elle permet en outre d'améliorer les performances car seules les bibliothèques nécessaires sont chargées.

- Lorsque vous ouvrez un document et que des relations sont rompues, il peut s'avérer nécessaire de charger des bibliothèques pour résoudre l'erreur, ce qui peut prendre un certain temps.
- Il est vivement recommandé d'identifier les paquets dont vous aurez besoin et de les sélectionner.

Par défaut, cette option est désactivée.

Tous les paquets

Permet de sélectionner tous les paquets.

Répertoire de référence pour les types



Répertoire de référence pour les types

Permet d'enregistrer le fichier CATGScript dans le répertoire indiqué dans la zone Répertoire de référence pour les types en vue d'une utilisation ultérieure (pour en savoir plus, voir PKT - Guide de l'utilisateur).

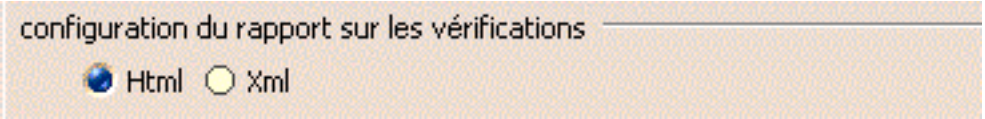
Par défaut, cette option n'est pas disponible.

Génération de rapport

Dans cette tâche, vous apprendrez à personnaliser les rapports générés par l'outil Global Check Analysis dans les ateliers Knowledge Advisor et Knowledge Expert. Elle traite des catégories d'options suivantes :

- Configuration du rapport sur les vérifications
- Fichier d'entrée XSL
- Contenu du rapport
- Répertoire de sortie
- Options HTML

Configuration du rapport sur les vérifications



Html

Permet de générer un rapport au format HTML.

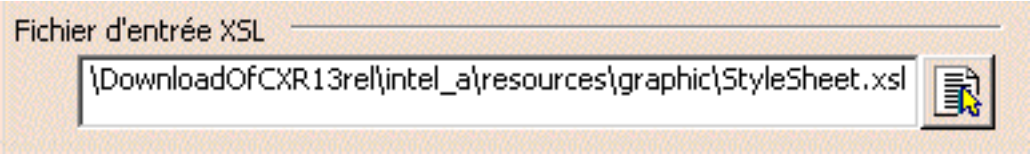
Xml

Permet de générer un rapport au format XML.

Par défaut, l'option HTML est activée.

Fichier d'entrée XSL

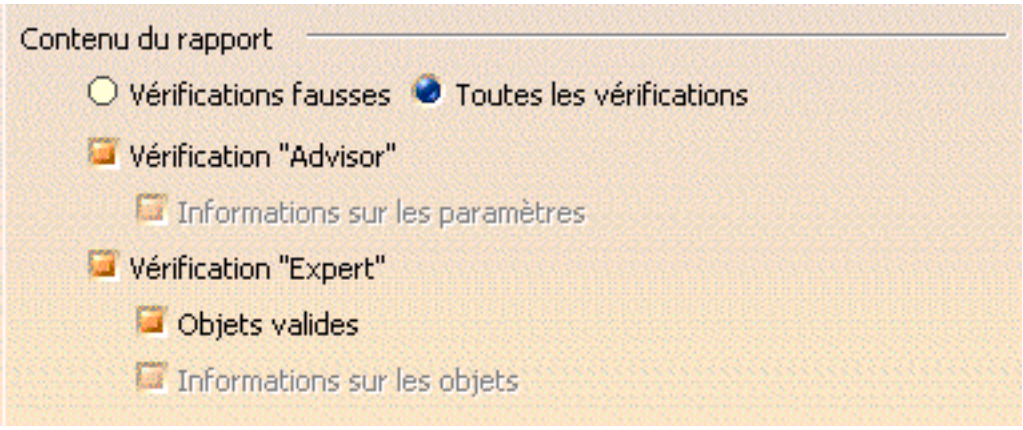
Cette option est uniquement disponible lorsque le paramètre de configuration XML est défini.



Fichier d'entrée XSL

Permet de sélectionner la feuille de style XSL appliquée au rapport XML. Le fichier StyleSheet.xsl est le fichier XSL par défaut, mais vous pouvez utiliser votre propre modèle.

Contenu du rapport



Vérifications fausses

Si cette option est cochée, le rapport généré contiendra uniquement les informations relatives aux vérifications fausses.

Par défaut, cette option est désactivée.

Toutes les vérifications

Si cette option est cochée, le rapport généré contiendra les informations relatives à toutes les vérifications qui figurent dans le document.

Par défaut, cette option est activée.

Vérification Advisor

Si cette option est cochée, le rapport généré contiendra les informations relatives à toutes les vérifications Knowledge Advisor qui figurent dans le document.

Par défaut, cette option est activée.

Informations sur les paramètres

Non disponible

Vérification Expert

Si cette option est cochée, le rapport généré contiendra les informations relatives à toutes les vérifications Knowledge Expert qui figurent dans le document.

Par défaut, cette option est activée.

Objets valides

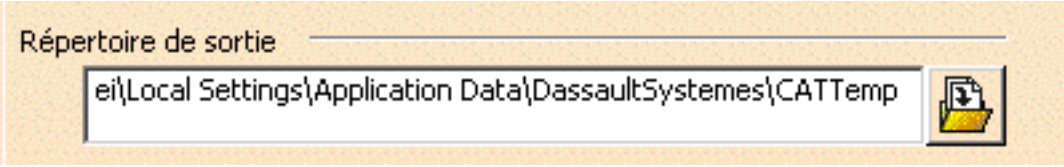
Si cette option est cochée, le rapport généré contiendra les informations relatives aux objets valides dans les vérifications expertes, ainsi que les informations relatives aux paramètres de ces objets (diamètre, profondeur, pas,...).

Par défaut, cette option est activée.

Informations sur les objets

Non disponible

Répertoire de sortie



Répertoire de sortie

Permet de sélectionner le répertoire de sortie contenant le rapport généré.

Par défaut, cette option est disponible.

Options HTML



Ouvrir le navigateur HTML dans une session CATIA

Cette option est disponible uniquement pour Windows. Elle permet à l'utilisateur de spécifier si le rapport sera ouvert dans une session *CATIA* (dans ce cas, l'option doit être cochée) ou s'il sera ouvert dans une session Internet Explorer (dans ce cas, l'option ne doit pas être cochée).

Il est fortement recommandé de ne pas utiliser ce rapport comme base pour les macros ou d'autres applications. Il est uniquement fourni à titre d'informations.

Par défaut, cette option est activée.

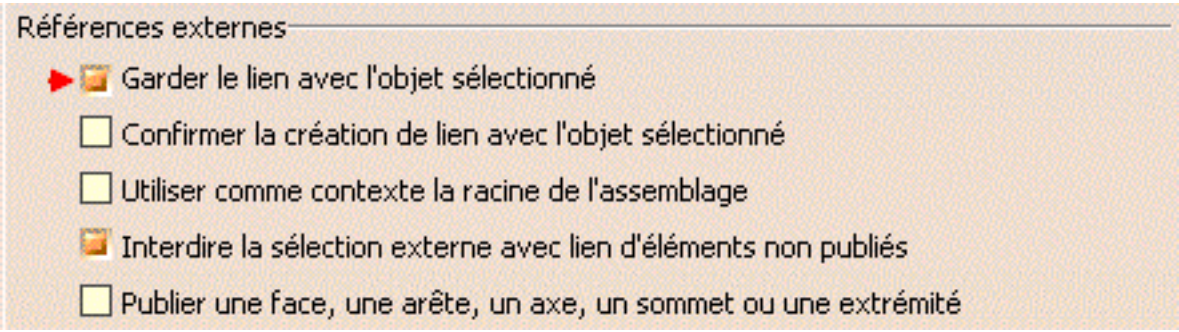
Part Infrastructure for Knowledgeware Applications



Cette section traite des options relatives aux éléments suivants :

- Les références externes : Garder le lien avec l'objet sélectionné
- L'affichage de l'arbre de spécifications

Options Général de Part Infrastructure

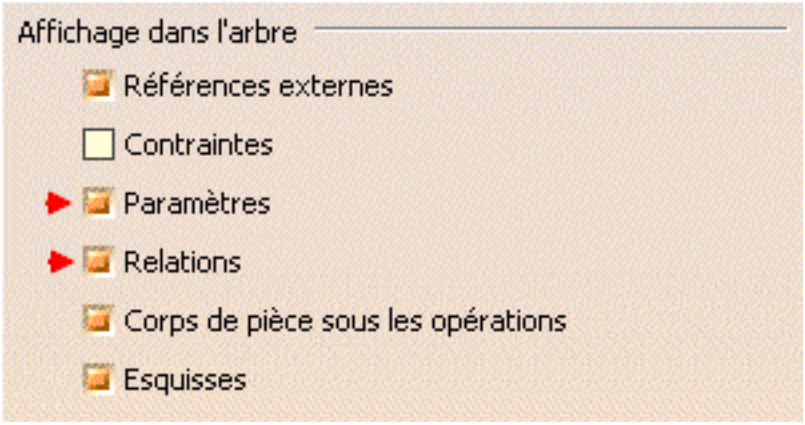


Garder le lien avec l'objet sélectionné

Activez cette option pour tirer parti de l'associativité.

Cliquez ici pour en savoir plus sur les options **Général** de Part Infrastructure.

Option Affichage de Part Infrastructure



Paramètres

Activez cette option pour afficher les paramètres en contexte CATProduct.

Par défaut, cette option est désactivée.

Relations

Activez cette option pour afficher les relations en contexte CATProduct.

Par défaut, cette option est désactivée.

Cliquez ici pour en savoir plus sur les options d'**affichage** de Part Infrastructure.

Référence

Les modules figurant dans la liste qui suit s'affichent dans l'explorateur lorsque vous spécifiez un corps de boucle.

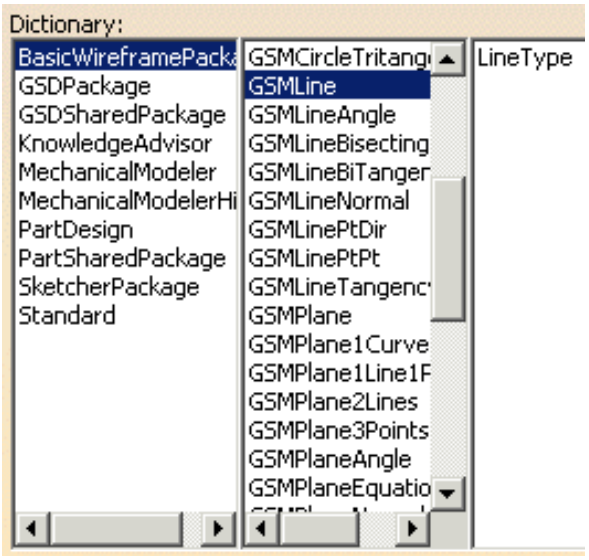
Wireframe - Version de base	Module GSD
Module GSD Shared	Knowledge Expert
Modeleur mécanique	Part Design
Module Part Shared	Module Standard

Wireframe - Version de base

GSMLine
GSMCircle
GSMPlane
GSMPoint

GSMLine





Définition :

Un objet GSMLine est une droite :

- généré par le produit Generative Shape Design,
- disponible dans le module BasicWireFrame.

Pour plus d'informations sur les droites, reportez-vous au document Generative Shape Design - Guide de l'utilisateur.

Attributs :

LineType

Une droite est définie par son type. L'attribut à utiliser est *LineType*. La syntaxe à utiliser est : **LineType = i**, où i correspond au type de droite à créer.

Le tableau d'équivalence suivant répertorie les types de droite que vous pouvez créer et le numéro à spécifier.

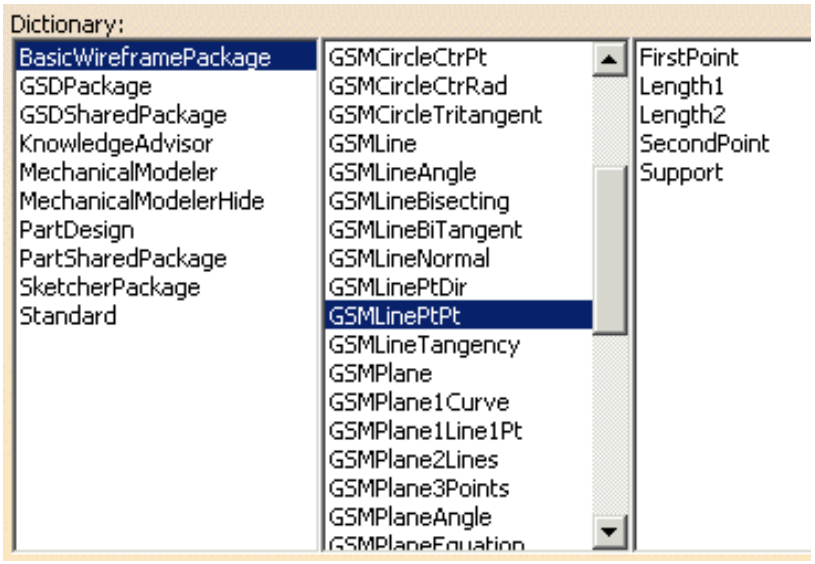
Type de ligne dans GSD	Type de ligne dans le module	Chiffre correspondant
Point - Point	GSMLinePtPt	0
Point-Direction	GSMLinePtDir	1
Angle - Courbe	GSMLineAngle	2
Tangente - Courbe	GSMLineTangency	3
Normale à une surface	GSMLineNormal	4
Intersection entre 2 plans	GSMLineBiTangent	5

Comme indiqué ci-dessus, vous pouvez créer sept sous-types différents de droite. Vous trouverez ci-après une description de chaque sous-type, ainsi que ses attributs et la syntaxe correspondante.

Droite Point - Point (*GSMLinePtpt*)

Le sous-type à utiliser dans ce cas est *GSMLinePtpt* qui définit les extrémités de la droite. Les attributs suivants sont disponibles pour ce sous-type :

- FirstPoint (composant)
- SecondPoint (composant)
- Support (composant)
- Length1 (longueur, facultatif pour les deux combinaisons)
- Length2 (longueur, facultatif pour les deux combinaisons)



Il est possible de combiner ces attributs de la façon suivante :

1ère combinaison	2ème combinaison
• *FirstPoint* qui est défini par la syntaxe suivante : FirstPoint = object: ..\..*theFirstPoint*;	• *FirstPoint* qui est défini par la syntaxe suivante : FirstPoint = object: ..\..*theFirstPoint*;
• *SecondPoint* qui est défini par la syntaxe suivante : SecondPoint = object: ..\..*theSecondPoint*;	• *SecondPoint* qui est défini par la syntaxe suivante : SecondPoint = object: ..\..*theSecondPoint*;
• Length1 qui est défini par la syntaxe suivante : Length1=200mm;	• Support
• Length2 qui est défini par la syntaxe suivante : Length2=150mm;	

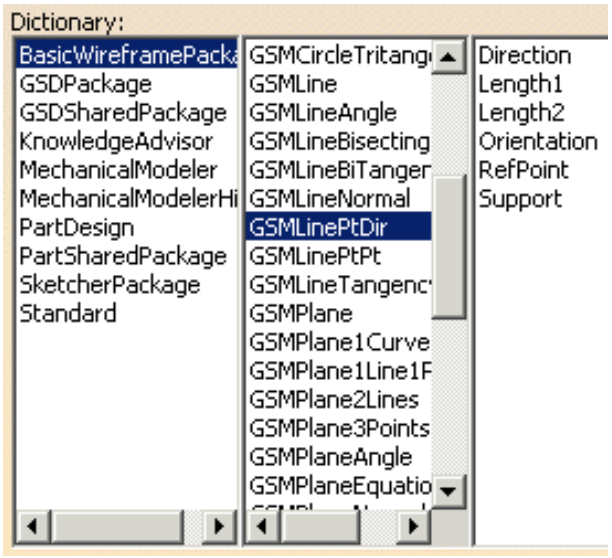
```
Line.1 isa GSMLine
{
    LineType = 0;
    TypeObject isa GSMLinePtPt
    {
     FirstPoint = object: Point.1;
     SecondPoint = object: Point.2;
     Length1 = 500mm; \\ optional
     Length2 = 435mm; \\ optional
    }
   }
```

```
Line.2 isa GSMLine
   {
    LineType = 0;
    TypeObject isa GSMLinePtPt
    {
     FirstPoint = object: ..\..\..\Point.1;
     SecondPoint = object: ..\..\..\Point.2;
     Support = object: ..\..\..\Extrude.1;
     Length1 = 50mm; \\ optional
     Length2 = 45mm; \\ optional
    }
   }
```

Point-Direction (*GSMLinePtDir*)

Le sous-type à utiliser dans ce cas est *GSMLinePtDir* qui définit la direction de la droite. Les attributs suivants sont disponibles pour ce sous-type :

- Length1
- Length2
- Direction
- Orientation
- RefPoint
- Support



Il est possible de combiner ces attributs de la façon suivante :

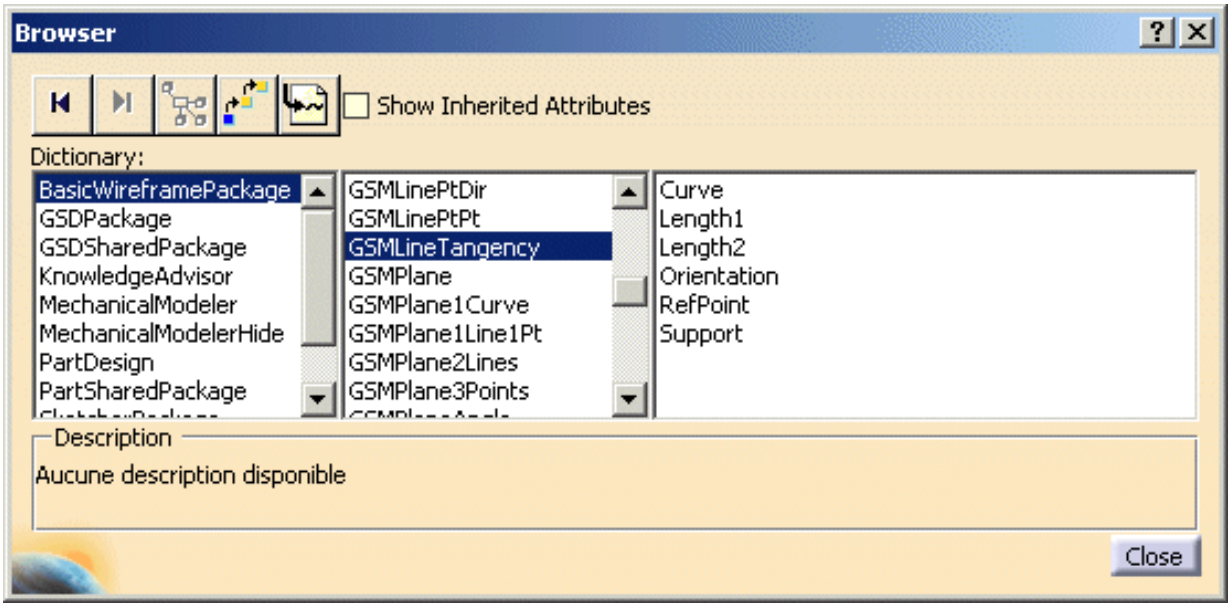
Combinaison

- Length1 qui est défini par la syntaxe suivante :
  `Length1 = 100mm;`
- Length2 qui est défini par la syntaxe suivante :
  `Length2 = 10mm;`
- Direction qui est défini par la syntaxe suivante :
  `Direction = object: ..\..\Plane.2;`
- *RefPoint* qui est défini par la syntaxe suivante :
  `RefPoint = object: ..\..\Point.2;`
- Support qui est défini par la syntaxe suivante :
  `SecondPoint = object: ..\..\'xy plane';`

Tangente - Courbe (*GSMLineTangency*)

Le sous-type à utiliser dans ce cas est *GSMLineTangency*. Les attributs suivants sont disponibles pour ce sous-type :

- Curve : courbe de référence utilisée pour définir la tangence.
- Length1
- Length2
- Orientation
- RefPoint : point de référence utilisé pour définir la tangence.
- Support



Il est possible de combiner ces attributs de la façon suivante :

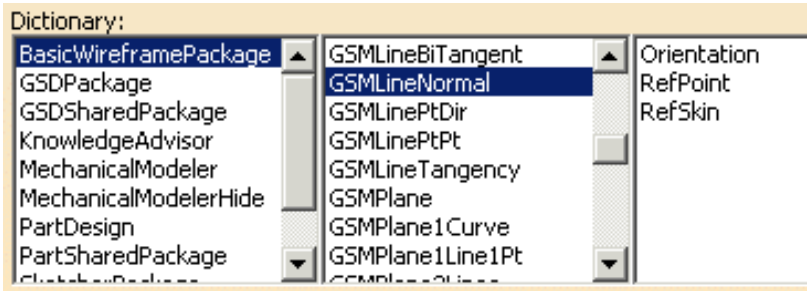
Combinaison

- Curve qui est défini par la syntaxe suivante :
  `Curve = object: ..\..\Spline.2;`
- Length1 qui est défini par la syntaxe suivante :
  `Length1 = 100mm;`
- Length2 qui est défini par la syntaxe suivante :
  `Length2 = 10mm;`
- *RefPoint* qui est défini par la syntaxe suivante :
  `RefPoint = object: ..\..\Point.2;`
- Support qui est défini par la syntaxe suivante :
  `SecondPoint = object: ..\..\'xy plane';`

Normal à une surface (*GSMLineNormal*)

Le sous-type à utiliser dans ce cas est *GSMLineNormal*. Les attributs suivants sont disponibles pour ce sous-type :

- Orientation
- RefPoint
- RefSkin



Il est possible de combiner ces attributs de la façon suivante :

Combinaison

- *RefPoint* qui est défini par la syntaxe suivante :
  RefPoint = object: ..\..\Point.2;
- Support qui est défini par la syntaxe suivante :
  RefSkin = object: ..\..\Extrude.1;

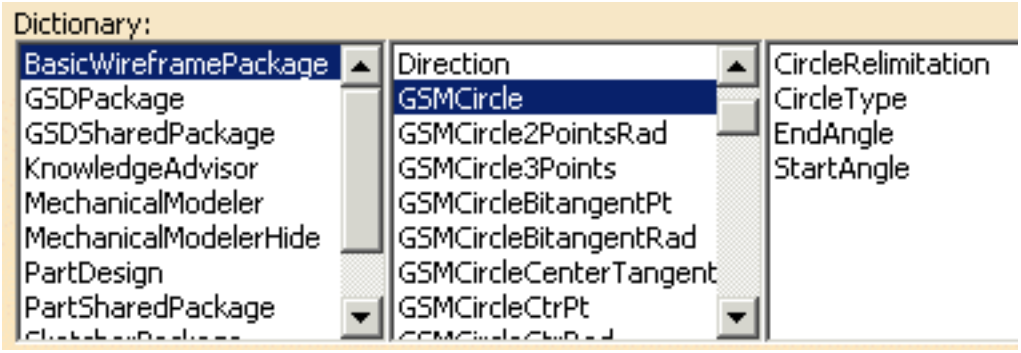
GSMCircle



Définition :

Un objet GSMCircle est un cercle :

- généré par le produit Generative Shape Design,
- disponible dans le module BasicWireFrame.



Pour plus d'informations sur les cercles, reportez-vous au document Generative Shape Design - Guide de l'utilisateur.

Attributs :

PointType

Un point est défini par les attributs suivants :

- *CircleType :* La syntaxe à utiliser est : **CircleType = i**, où i correspond au type de droite à créer.
- *CircleRelimitation :* La syntaxe à utiliser est **CircleRelimitation =**.
- *EndAngle :* La syntaxe à utiliser est **EndAngle =** xxxdeg.
- *StartAngle :* La syntaxe à utiliser est **StartAngle =**xxxdeg.

Le tableau suivant répertorie les types de cercle que vous pouvez créer et le numéro à spécifier.

Type de plan dans GSD	Type de plan dans le module	Chiffre correspondant

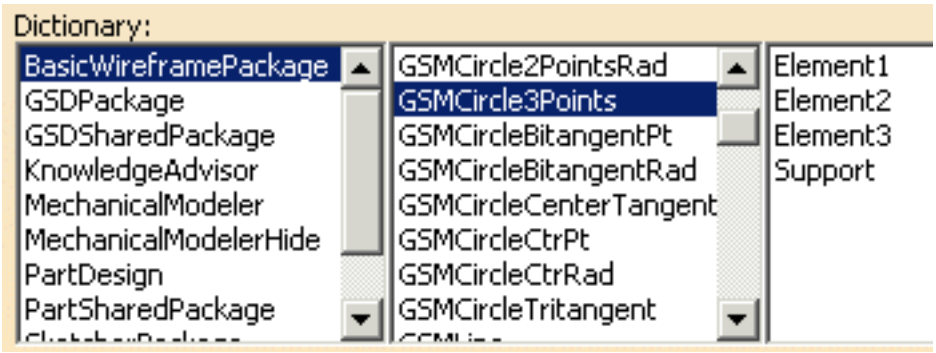
Trois points	GSMPCircle3Points	3
Centre - Rayon	GSMCircleCtrRad	0
Centre - Point	GSMCircleCtrPt	1

Comme indiqué ci-dessus, vous pouvez créer trois sous-types différents de cercle. Vous trouverez ci-après une description de chaque sous-type, ainsi que ses attributs et la syntaxe correspondante.

Trois points (*GSMCircle3Points*)

Le sous-type à utiliser dans ce cas est *GSMCircle3Points* qui permet de créer un cercle passant par trois points. Les attributs suivants sont disponibles pour ce sous-type :

- Element1 : premier point
- Element2 : deuxième point
- Element3 : troisième point
- Support : surface support sur laquelle sera projeté le cercle (facultatif)



Il est possible de combiner ces attributs de la façon suivante :

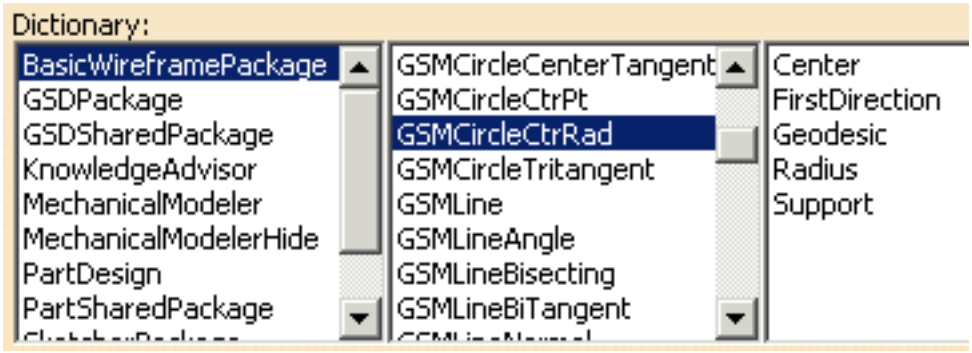
Combinaison

- Element1 qui est défini par la syntaxe suivante :
  `Element1 = object: ..\Point.1;`
- Element2 qui est défini par la syntaxe suivante :
  `Element2 = object: ..\Point.2;`
- Element3 qui est défini par la syntaxe suivante :
  `Element3 = object: ..\Point.3;`
- Support qui est défini par la syntaxe suivante :
  `Support = object: ..\Extrude.1;`

Centre - Rayon (*GSMCircleCtrRad*)

Le sous-type à utiliser dans ce cas est *GSMCircleCtrRad* qui permet de créer un cercle en indiquant son centre et son rayon. Les attributs suivants sont disponibles pour ce sous-type :

- Center : point qui sera le centre du cercle.
- FirstDirection
- Geodesic
- Radius : rayon du cercle.
- Support : plan ou surface support sur lequel créer le cercle.



Il est possible de combiner ces attributs de la façon suivante :

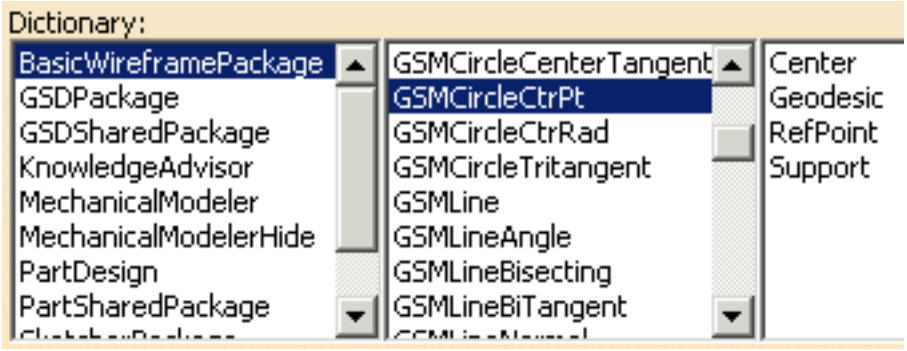
Combinaison

- Center qui est défini par la syntaxe suivante :
  `Center = object: ..\Point.1;`
- `Radius` qui est défini par la syntaxe suivante :
  `Radius = 120mm;`
- `Support` qui est défini par la syntaxe suivante :
  `Support = object: ..\Extrude.1;`

Centre - Point (*GSMCircleCtrPt*)

Le sous-type à utiliser dans ce cas est *GSMCircleCtrPt* qui permet de créer un cercle en indiquant son centre et un point. Les attributs suivants sont disponibles pour ce sous-type :

- Center : point qui sera le centre du cercle.
- Geodesic : Courbe.
- RefPoint : deuxième point utilisé pour créer le cercle.
- Support : plan ou surface support sur lequel créer le cercle.

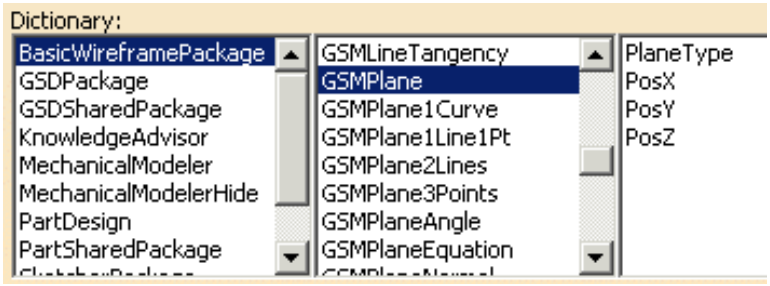


Combinaison

- Center qui est défini par la syntaxe suivante :
  Center = object: ..\Point.1;
- RefPoint qui est défini par la syntaxe suivante :
  RefPoint = object: ..\Point.1;
- Support qui est défini par la syntaxe suivante :
  Support = object: ..\Extrude.1;

GSMPlane

Définition :



Un objet GSMPlane est un plan :

- généré par le produit Generative Shape Design,
- disponible dans le module BasicWireFrame.

Pour plus d'informations sur les plans, reportez-vous au document Generative Shape Design - Guide de l'utilisateur.

Attributs :

PlaneType

Un plan est défini par son type. L'attribut à utiliser est *PlaneType*. La syntaxe à utiliser est : **PlaneType = i**, où i correspond au type de plan à créer.

Le tableau suivant répertorie les types de plan que vous pouvez créer et le numéro à spécifier.

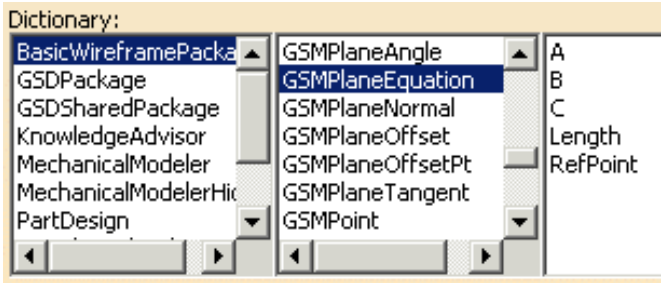
Type de plan dans GSD	Type de plan dans le module	Chiffre correspondant
Equation	GSMPlaneEquation	0
Par 3 points	GSMPlane3Points	1
Par 2 droites	GSMPlane2Lines	2
Par un point et une droite	GSMPlane1line1Pt	3
Normal à une courbe	GSMPlane1Curve	4
Tangent à une surface	GSMPlaneTangent	5
Normal à un plan	GSMPlaneNormal	6

Comme indiqué ci-dessus, vous pouvez créer sept sous-types différents de plan. Vous trouverez ci-après une description de chaque sous-type, ainsi que ses attributs et la syntaxe correspondante.

Equation (*GSMPlaneEquation*)

Le sous-type à utiliser dans ce cas est *GSMPlaneEquation* qui permet de créer un plan passant par une équation. Les attributs suivants sont disponibles pour ce sous-type :

- A (premier élément de l'équation)
- B (deuxième élément de l'équation)
- C (troisième élément de l'équation)
- Length
- RefPoint (point utilisé pour positionner le plan via ce point)



Il est possible de combiner ces attributs de la façon suivante :

1ère combinaison	2ème combinaison
• A qui est défini par la syntaxe suivante : A=31; //une valeur est obligatoire • B qui est défini par la syntaxe suivante : B=-47; //une valeur est obligatoire • C qui est défini par la syntaxe suivante : C=-24; //une valeur est obligatoire • Length : permet à l'utilisateur d'indiquer la longueur requise. Il est défini par la syntaxe suivante : Length=24mm	• A qui est défini par la syntaxe suivante : A=31; //une valeur est obligatoire • B qui est défini par la syntaxe suivante : B=-47; //une valeur est obligatoire • C qui est défini par la syntaxe suivante : C=-24; //une valeur est obligatoire • RefPoint qui est défini par la syntaxe suivante : RefPOint = object: ..\Point ;

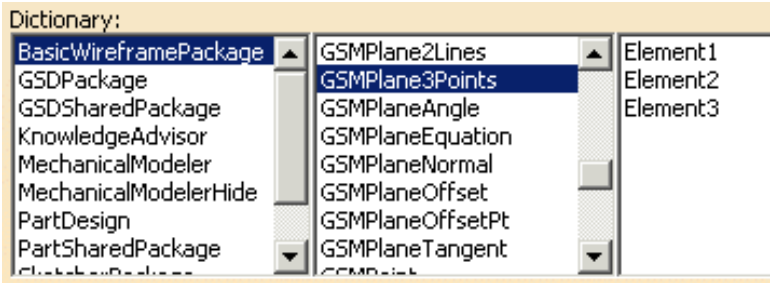
```
Plane0.1 isa GSMPlane
{
   PlaneType = 0;
   TypeObject isa GSMPlaneEquation
    {
      A = 15;
      B = -12;
      C = 31;
      Length = 24mm;
    }
}
```

```
Plane0.2 isa GSMPlane
{
   PlaneType = 0;
   TypeObject isa GSMPlaneEquation
    {
      A = 31;
      B = -47;
      C = -24;
      RefPoint = object: ..\ConstrBody\Point.5;
    }
}
```

Par 3 points (*GSMPlane3Points*)

Le sous-type à utiliser dans ce cas est *GSMPlane3Points* qui permet de créer un plan passant par 3 points. Les attributs suivants sont disponibles pour ce sous-type :

- Element1 (premier point)
- Element2 (deuxième point)
- Element3 (troisième point)



Il est possible de combiner ces attributs de la façon suivante :

Combinaison

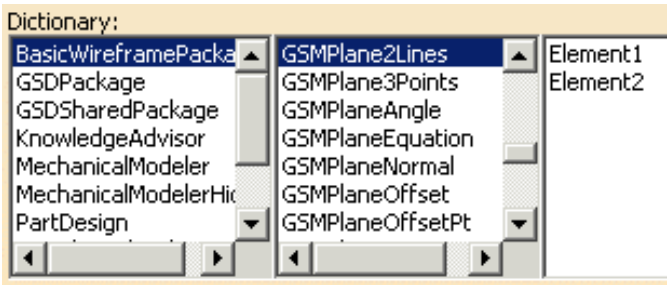
- Element1 qui est défini par la syntaxe suivante :
  Element1 = object: ..\Point.1;
- Element2 qui est défini par la syntaxe suivante :
  Element2 = object: ..\Point.2;
- Element3 qui est défini par la syntaxe suivante :
  Element3 = object: ..\Point.3;

```
Plane1 isa GSMPlane
{
   PlaneType = 1;
   TypeObject isa GSMPlane3Points
            {
      Element1 = object: ..\Point.1;
      Element2 = object: ..\Point.5;
      Element3 = object: ..\Point.8;
            }
}
```

Par 2 droites (*GSMPlane2Lines*)

Le sous-type à utiliser dans ce cas est *GSMPlane2Lines* qui permet de créer un plan passant par deux droites. Les attributs suivants sont disponibles pour ce sous-type :

- Element1 (première droite)
- Element2 (deuxième droite)



Combinaison

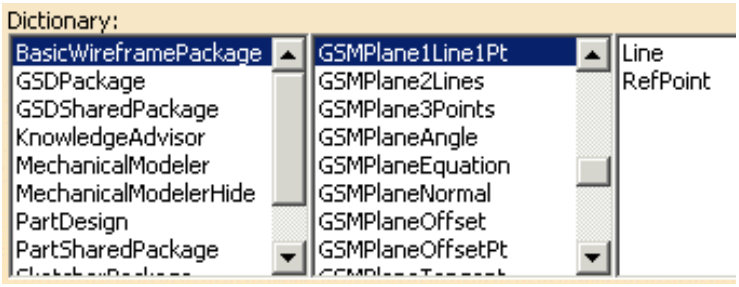
- Element1 qui est défini par la syntaxe suivante :
  Element1 = object: ..\Line.1;
- Element2 qui est défini par la syntaxe suivante :
  Element2 = object: ..\Line.2;

```
Plane2 isa GSMPlane
{
   PlaneType = 2;
   TypeObject isa GSMPlane2Lines
            {
                Element1 = object: ..\Line.1;
                Element2 = object: ..\Line.2;
            }
}
```

Par un point et une droite (*GSMPlane1line1Pt*)

Le sous-type à utiliser dans ce cas est *GSMPlane1Line1Pt* qui permet de créer un plan passant par une droite et un point. Les attributs suivants sont disponibles pour ce sous-type :

- Line : droite utilisée pour créer le plan.
- RefPoint : point utilisé pour créer le plan.



Utilisez les attributs comme suit :

Combinaison

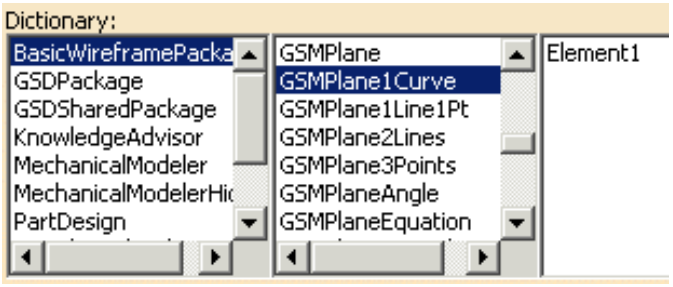
- Line qui est défini par la syntaxe suivante :
  `Line = object: ..\Line.1;`
- RefPoint qui est défini par la syntaxe suivante :
  `RefPoint = object: ..\Point.2;`

```
Plane3 isa GSMPlane
{
   PlaneType = 3;
   TypeObject isa GSMPlane1line1Pt
            {
                Line = object: ..\Line.1;
                RefPoint = object: ..\Point.13;
            }
}
```

Normal à une courbe (*GSMPlane1Curve*)

Le sous-type à utiliser dans ce cas est *GSMPlane1Curve* qui permet de créer un plan normal à une courbe en un point spécifié.

- Element1: Line



Utilisez cet attribut comme suit :

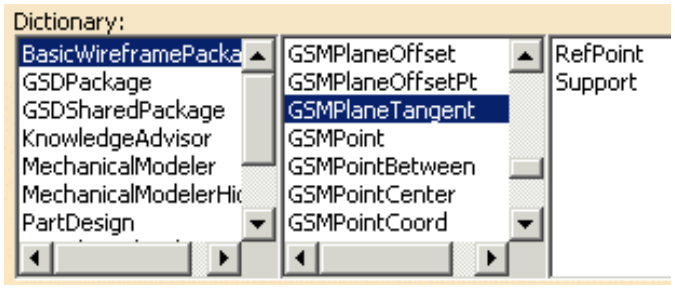
Combinaison

- Line qui est défini par la syntaxe suivante :
  `Line = object: ..\Spline.1;`

Tangent à une surface (*GSMPlaneTangent*)

Le sous-type à utiliser dans ce cas est *GSMPlaneTangent* qui permet de créer un plan tangent à une surface en un point spécifié. Les attributs suivants sont disponibles pour ce sous-type :

- RefPoint (Point)
- Support (Surface)



Utilisez les attributs comme suit :

Combinaison

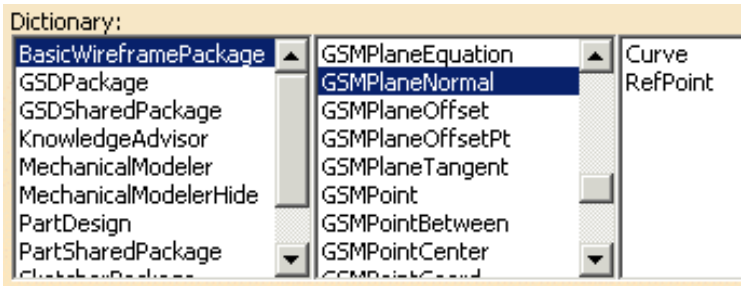
- Support qui est défini par la syntaxe suivante :
  `Support = object: ..\Spline.1;`
- RefPoint qui est défini par la syntaxe suivante :
  `RefPoint = object: ..\Point.4;`

```
Plane5 isa GSMPlane
{
   PlaneType = 5;
   TypeObject isa GSMPlaneTangent
              {
                  Support = object: ..\Extrude.1;
                  RefPoint = object: ..\Point.4;
              }
}
```

Normal à un plan (GSMPlaneNormal)

Le sous-type à utiliser dans ce cas est *GSMPlaneNormal*. Les attributs suivants sont disponibles pour ce sous-type :

- Curve : courbe de référence utilisée pour créer le plan.
- RefPoint : point de référence utilisé pour créer le plan.



Utilisez les attributs comme suit :

Combinaison

- Curve qui est défini par la syntaxe suivante :
  `Curve = object: ..\Spline.1;`
- RefPoint qui est défini par la syntaxe suivante :
  `RefPoint = object: ..\Point.4;`

```
Plane6 isa GSMPlane
{
   PlaneType = 6;
   TypeObject isa GSMPlaneNormal
             {
                 Curve = object: ..\Spline.2;
                 RefPoint = object: ..\Point.13;
             }
}
```

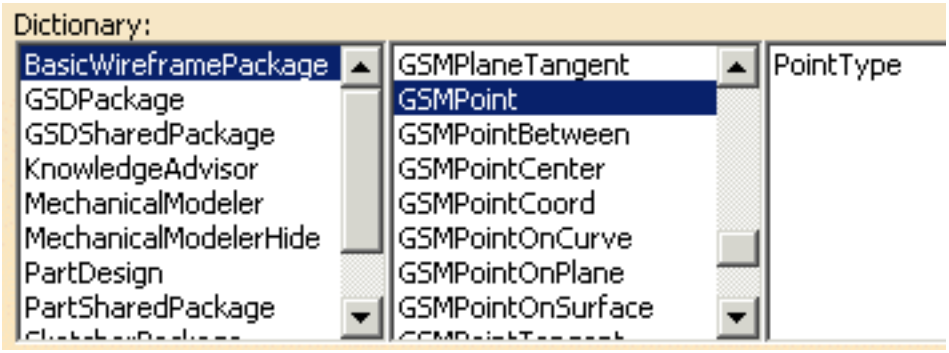
GSMPoint



Définition :

Un objet GSMPoint est un point :

- généré par le produit Generative Shape Design,
- disponible dans le module BasicWireFrame.



Pour plus d'informations sur les points, reportez-vous au document Generative Shape Design - Guide de l'utilisateur.

Attributs :

PointType

Un point est défini par son type. L'attribut à utiliser est *PointType*. La syntaxe à utiliser est : **PointType = i**, où i correspond au type de point à créer.

Le tableau suivant répertorie les types de point que vous pouvez créer et le numéro à spécifier.

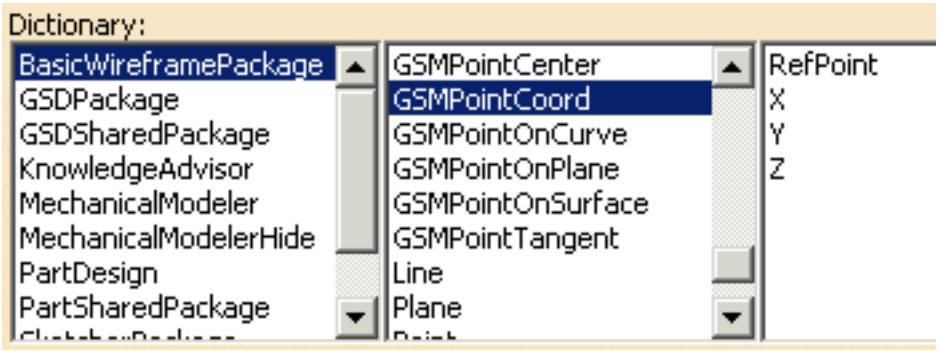
Type de plan dans GSD	Type de plan dans le module	Chiffre correspondant
Coordonnées	GSMPointCoord	0
Sur une surface	GSMPointOnSurface	1
Sur une courbe	GSMPointOnCurve	2
Sur un plan	GSMPointOnPlane	3
Centre du cercle	GSMPointCenter	4

Comme indiqué ci-dessus, vous pouvez créer cinq sous-types différents de point. Vous trouverez ci-après une description de chaque sous-type, ainsi que ses attributs et la syntaxe correspondante.

Coordonnées (*GSMPointCoord*)

Le sous-type à utiliser dans ce cas est *GSMPointCoord* qui permet de créer un point de coordonnée. Les attributs suivants sont disponibles pour ce sous-type :

- RefPoint (point de référence, facultatif). S'il est défini, x, y et z sont indiqués sur une marque dont l'origine est ce point de référence.
- X (première coordonnée)
- Y (deuxième coordonnée)
- Z (troisième coordonnée)



Il est possible de combiner ces attributs de la façon suivante :

Combinaison

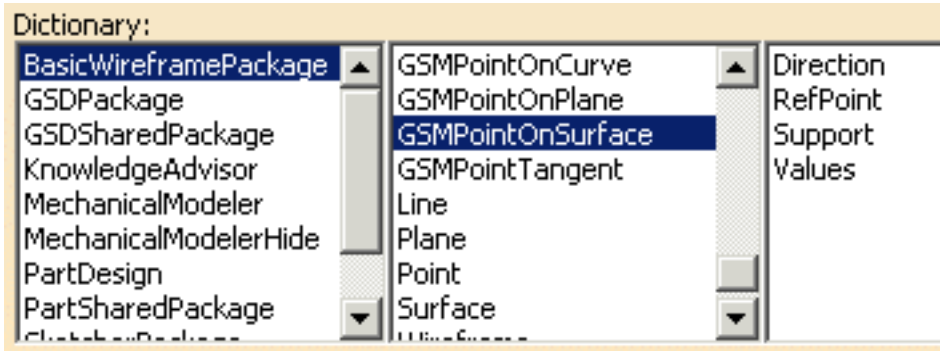
- RefPoint (point de référence, facultatif)
- X qui est défini par la syntaxe suivante :
  `X = 10mm;`
- Y qui est défini par la syntaxe suivante :
  `Y = 10mm;`
- Y qui est défini par la syntaxe suivante :
  `Z = 10mm;`

```
Point0.2 isa GSMPoint
    {
      PointType = 0;
      TypeObject isa GSMPointCoord
      {
        RefPoint = object: ..\Construction_Body\GSMPoint.21;
        X= 12mm;
        Y= 15mm;
        Z= 18mm;
      }
    }
```

Sur la surface (*GSMPointOnSurface*)

Le sous-type à utiliser dans ce cas est *GSMPointOnSurface* qui permet de créer un point sur un plan. Les attributs suivants sont disponibles pour ce sous-type :

- Direction : élément dont l'orientation sert de direction de référence ou un plan dont la normale sert de direction de référence
- RefPoint : point de référence. Par défaut, le point central de la surface sert de référence.
- Support : surface sur laquelle doit être créé le point.
- Values : distance le long de la direction de référence pour afficher un point.



Il est possible de combiner ces attributs de la façon suivante :

Combinaison

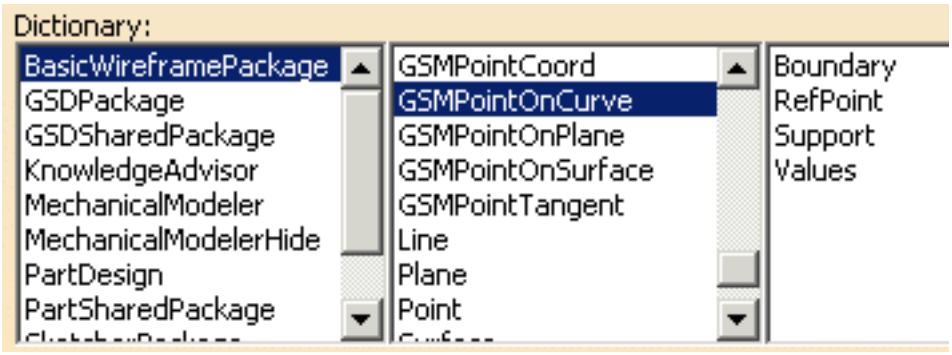
- Direction qui est défini par la syntaxe suivante :
  `Direction = object: ..\Line.1;`
- Support qui est défini par la syntaxe suivante :
  `Support= object: ..\Extrude.1;`
- Values qui est défini par la syntaxe suivante :
  `Values = 12mm;`

```
Point1 isa GSMPoint
{
  PointType = 1;
  TypeObject isa GSMPointOnSurface
  {
  Direction = object: ..\Construction_Body\Line.4;
  Support = object: ..\Construction_Body\GSMExtrude.1;
  Values = 25mm;
  }
}
```

Sur la courbe (*GSMPointOnCurve*)

Le sous-type à utiliser dans ce cas est *GSMPointOnCurve* qui permet de créer un point sur une courbe. Les attributs suivants sont disponibles pour ce sous-type :

- Boundary : non disponible.
- RefPoint : point de référence. S'il n'est pas défini, il s'agit de l'extrémité de la courbe.
- Support : courbe
- Values : distance entre le point de référence et ce point.



Combinaison

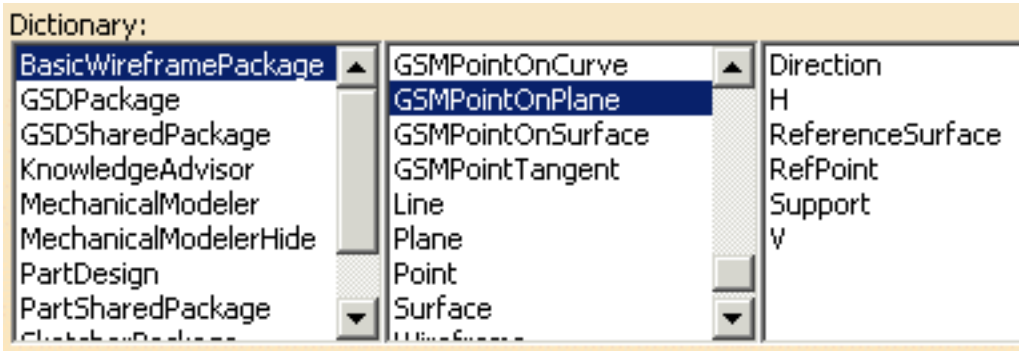
- Refpoint qui est défini par la syntaxe suivante :
  RefPoint= object: ..\Point.1;
- Support qui est défini par la syntaxe suivante :
  Support = object: ..\Line.1;
- Values qui est défini par la syntaxe suivante :
  Values = 12mm;

```
Point2 isa GSMPoint
{
  PointType = 2;
  TypeObject isa GSMPointOnCurve
  {
  Support = object: ..\Construction_Body\GSMLine.3;
  Values = 125mm;
  }
}
```

Sur le plan (*GSMPointOnPlane*)

Le sous-type à utiliser dans ce cas est *GSMPointOnPlane*. Il crée un point sur un plan. Les attributs suivants sont disponibles pour ce sous-type :

- Direction (facultatif). Lorsqu'il est défini, indique la direction.
- H : vecteur.
- RefPoint : point utilisé pour définir une référence pour le calcul des coordonnées du plan.
- Support : plan sur lequel sera créé le point.
- V : vecteur.



Utilisez les attributs comme suit :

Combinaison

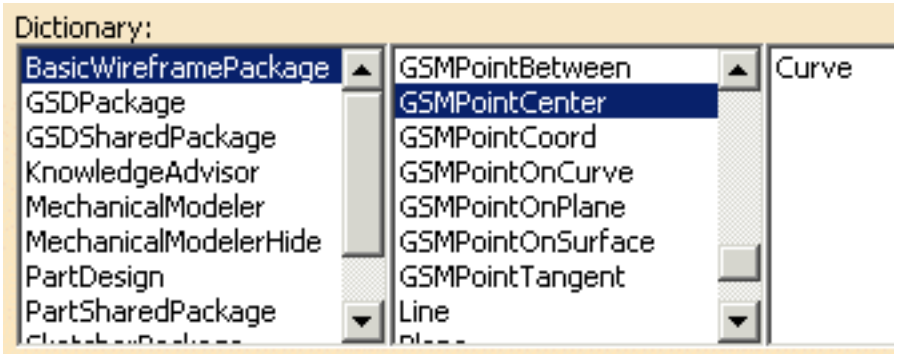
- Direction qui est défini par la syntaxe suivante :
  `Direction = object: ..\Line.1;`
- H qui est défini par la syntaxe suivante :
  `H = 150mm;`
- RefPoint qui est défini par la syntaxe suivante :
  `RefPoint= object: ..\Point.1;`
- Support qui est défini par la syntaxe suivante :
  `Support = object: 'xy plane'`
- V qui est défini par la syntaxe suivante :
  `V = 150mm;`

```
Point3 isa GSMPoint
{
  PointType = 3;
  TypeObject isa GSMPointOnPlane
  {
  Support = object: ..\..\..\`xy plane`;
  H = 150mm;
  V = 120mm;
  }
}
```

Centre du cercle (GSMPointCenter)

Le sous-type à utiliser dans ce cas est *GSMPointCenter* qui permet de définir le centre d'un cercle.

- Curve : cercle, arc de cercle ou ellipse.



Utilisez cet attribut comme suit :

Combinaison

- Curve qui est défini par la syntaxe suivante :
  Curve = object: ..\Extrude.1;

```
Point4 isa GSMPoint
{
  PointType = 4;
  TypeObject isa GSMPointCenter
  {
    Curve = object: ..\GSMExtrude.2;
            }
}
```

Module Part Design

Le tableau suivant répertorie les types contenus dans le module Part Design.

Boîte	Chanfrein	Cône
Trou lamé	Trou lamé chanfreiné	Trou chanfreiné
Cylindre	Trou	Extrusion
Pocket	*RemoveFace*	*ReplaceFace*
Révolution	Coque	SimpleHole
SoldCombine	*Coupe*	TaperedHole
Epaisseur	Surface épaisse	Torus

Boîte

Définition :

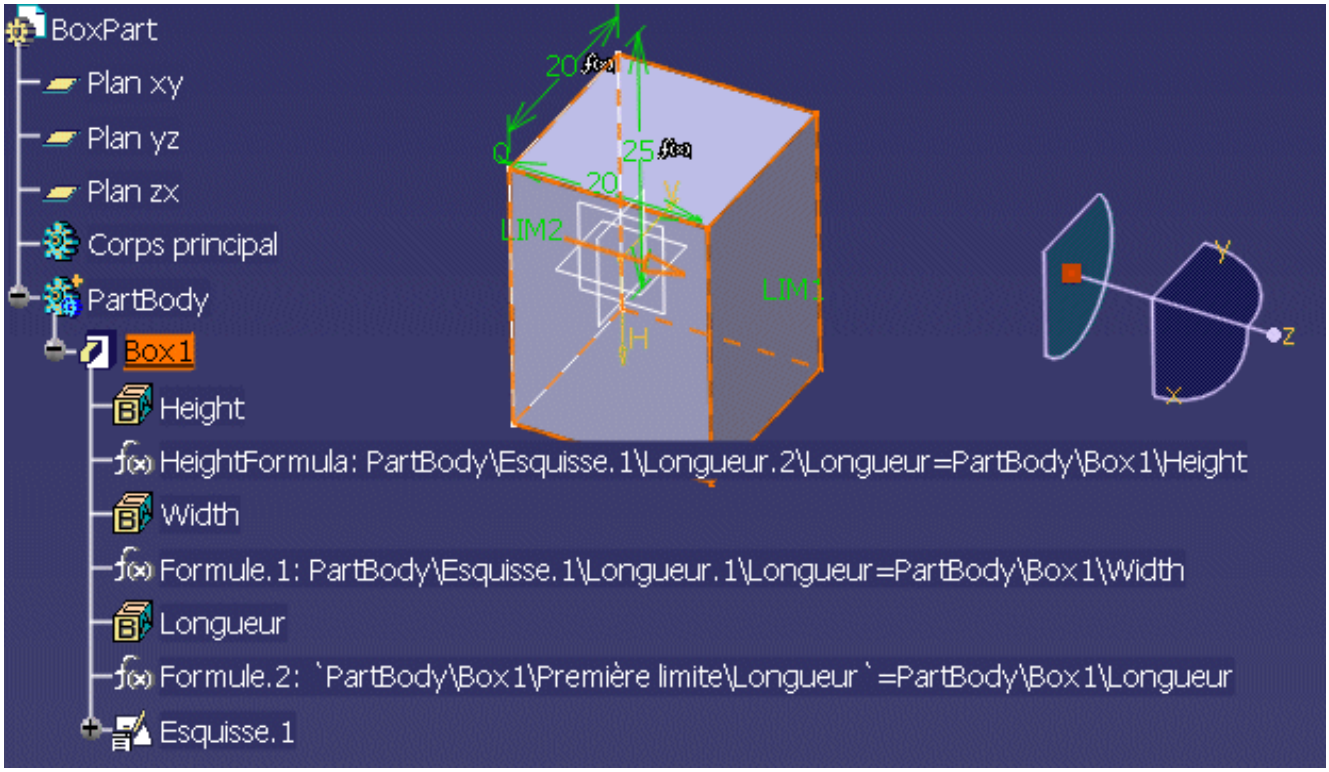
Une boîte est une extrusion produite à partir d'une esquisse rectangulaire.

Attributs :

Une boîte est définie par les attributs suivants :

- *Length* (longueur), qui indique la limite de l'extrusion. La syntaxe à utiliser est **Length = 10mm**.
- *Width (largeur)*, qui indique la largeur de l'extrusion. La syntaxe à utiliser est **Width = 20mm**.
- *Height* (hauteur), qui indique la hauteur de l'extrusion. La syntaxe à utiliser est **Height = 12mm**.

```
MyBox isa CATPart
 {
 BoxPart isa Part
  {
   PartBody isa BodyFeature
    {
    // Crée une boîte
    Box1 isa Box
     {
     // Spécifie les propriétés de la boîte
     Width = 20.0 mm ;
     Height = 25.0 mm ;
     Length = 15.0 mm ;
     }
    }
  }
```

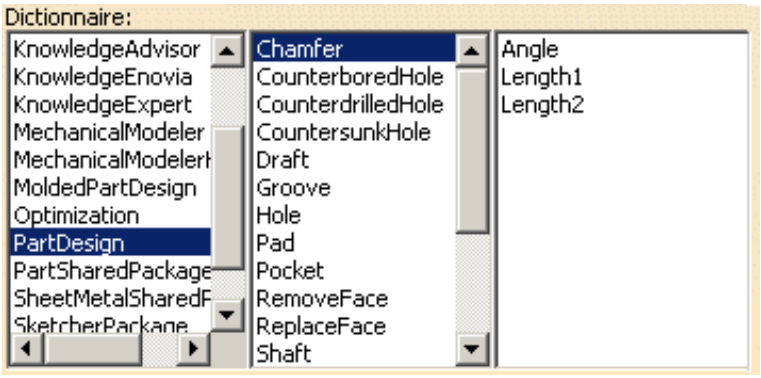


```
}
```

Chanfrein

Définition :

Découpe angulaire réalisée dans l'épaisseur d'un composant, produisant une arête en biais.



Attributs :

Un chanfrein est défini par les attributs suivants :

- *Angle*. La syntaxe à utiliser est **Angle = 20 deg**.
- *Length1*. La syntaxe à utiliser est **Length1 = 5 mm**.
- *Length2*. La syntaxe à utiliser est **Length2 = 5 mm**.

Remarques importantes :

- Un chanfrein a un attribut Length2 qui indique la longueur par défaut du chanfrein. Cet attribut ne doit pas être manipulé dans un script.

Pour définir un chanfrein dans un script, vérifiez qu'une pièce est ouverte, puis procédez comme suit :

1. Créez un trou à l'aide de la commande isa

   ```
   Chamfer1 isa Chamfer ( ) { }
   ```

2. Avec le bouton droit, cliquez n'importe où entre les parenthèses et sélectionnez la commande "Insérer arête" ou "Insérer surface" du menu contextuel. Puis, dans la zone géométrique, sélectionnez une arête ou une surface à chanfreiner.

```
MyBox isa
CATPart
  {
   BoxPart isa
Part
    {
     PartBody
isa BodyFeature
       {
        // Crée
une boîte
        Box1
isa Box
         {

Width = 20.0
mm ;

Height = 25.0
mm ;

Length = 15.0
```

```
mm ;
        }
        // Crée un chanfrein
        // La définition de l'arête doit être capturée
        // dans la zone géométrique
        // Utilise la commande Insérer arête du
        // menu contextuel

Chamfer1 isa Chamfer (Edge Definition)
        {

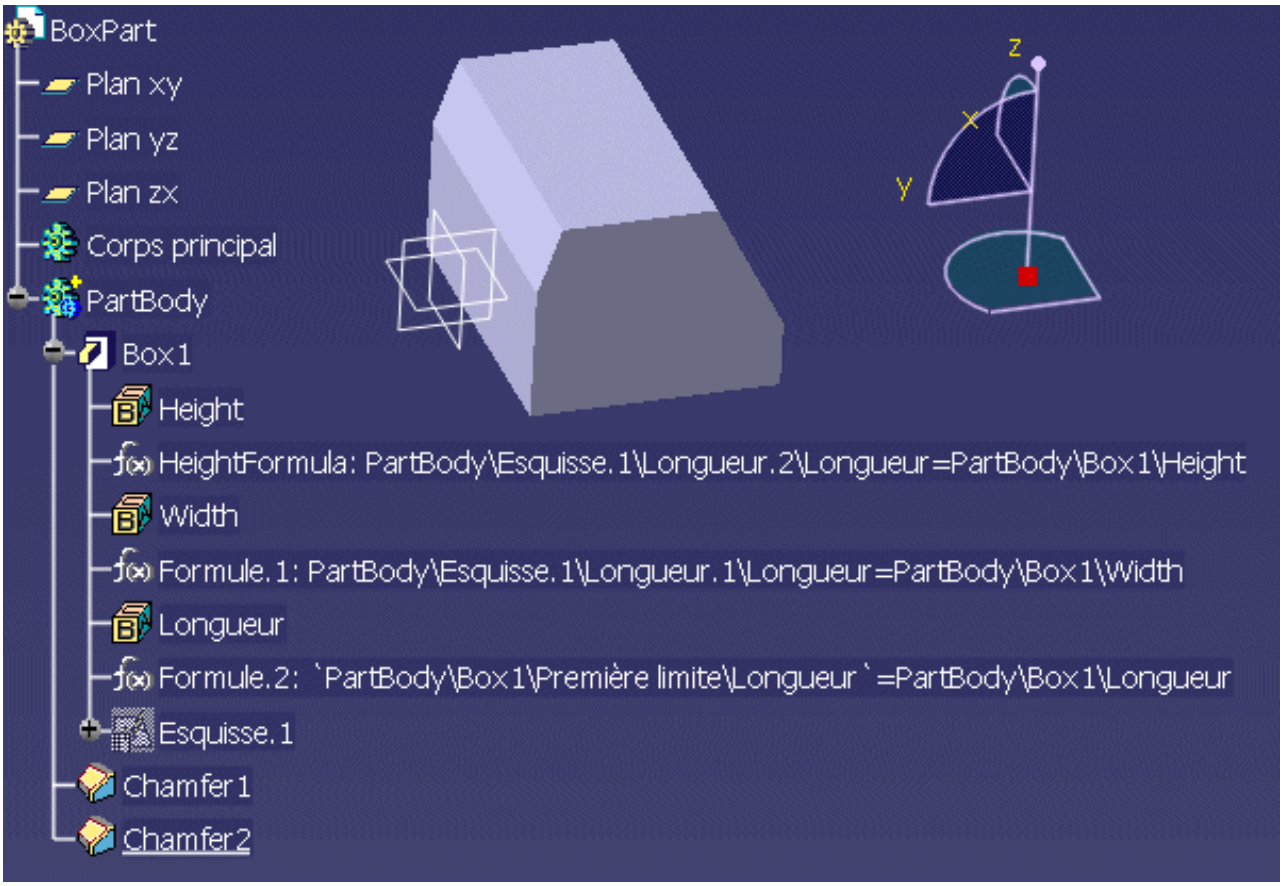
Angle = 20 deg;

Length1 = 5 mm ;
        }

Chamfer2 isa Chamfer (Edge Definition)
        {

Angle = 30 deg;

Length1 = 10 mm ;
        }
      }
    }
  }
```



Cône

Définition :

Un cône est un axe créé par rotation d'une esquisse triangulaire.

Attributs :

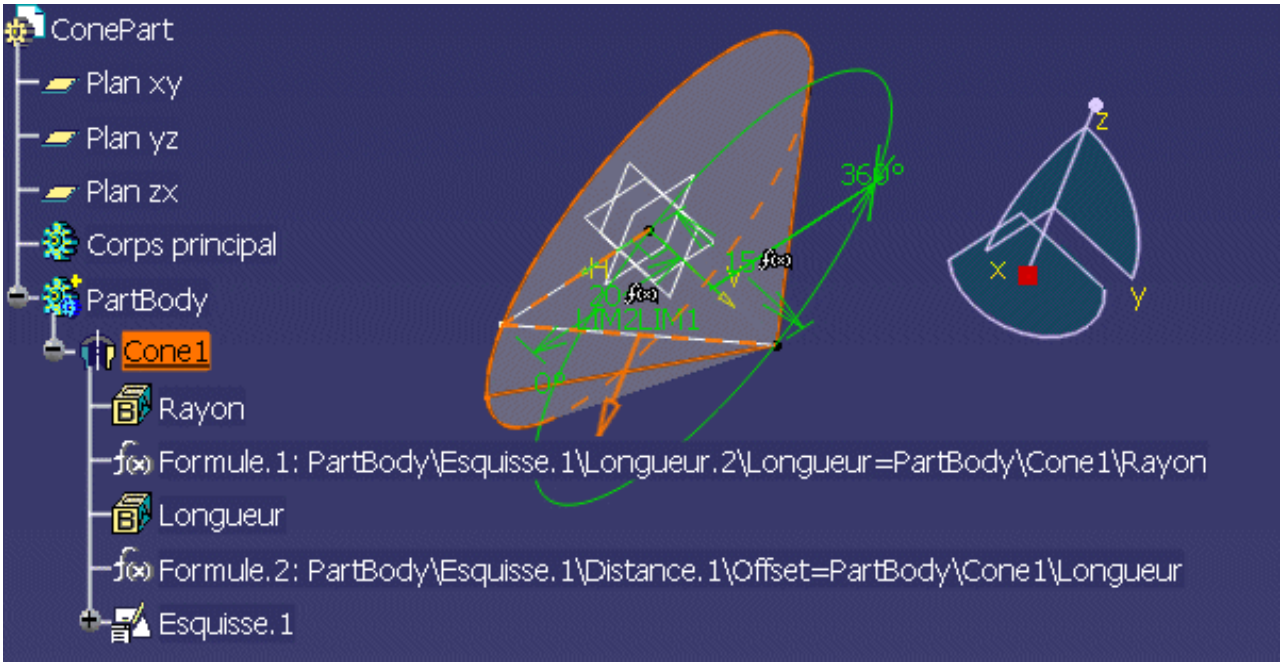
Un cône est défini par les attributs suivants :

- Length. La syntaxe à utiliser est **Length = 15.0 mm**.
- Radius. La syntaxe à utiliser est **Radius = 20.0 mm**.

```
MyCone isa
CATPart
  {
   ConePart isa
Part
     {
      PartBody
isa BodyFeature
       {
        // Crée
un cône
         Cone1
isa Cone
           {

Radius = 20.0
mm ;

Length = 15.0
mm ;
           }
       }
     }
 }
```

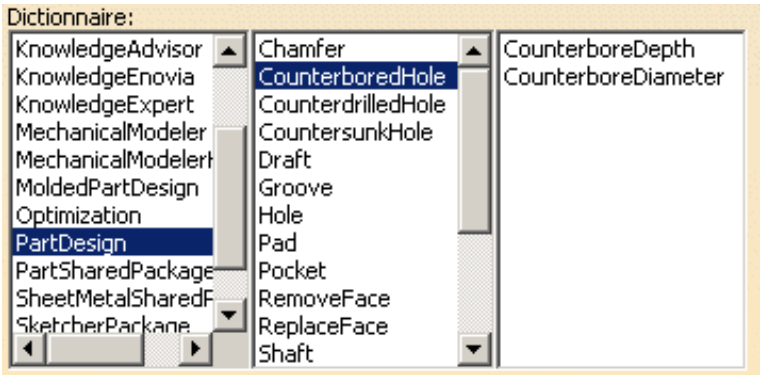


Trou lamé

Définition :

Entité mécanique de type trou créé lorsque vous cliquez sur l'icône  dans l'atelier Part Design. Pour plus d'informations, reportez-vous au document *Part Design - Guide de l'utilisateur*.





Attributs :

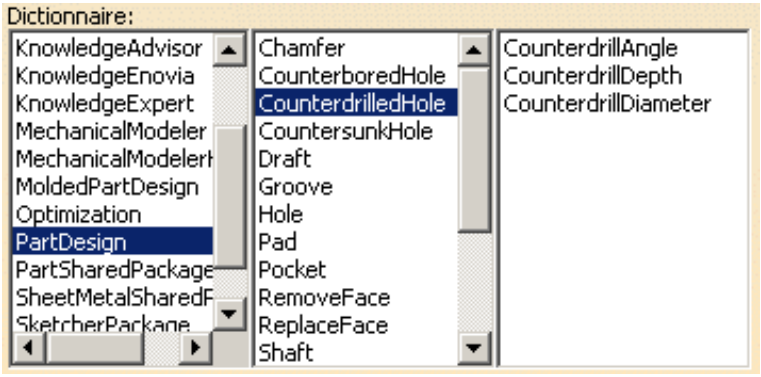
Un trou lamé est défini par les attributs suivants :

- CounterboreDepth. La syntaxe à utiliser est **CounterboreDepth = 12mm**.
- CounterboreDiameter : La syntaxe à utiliser est **CounterboreDiameter =15mm**.

Trou lamé chanfreiné

Définition :

Entité mécanique de type trou créé lorsque vous cliquez sur l'icône dans l'atelier Part Design. Pour plus d'informations, reportez-vous au document *Part Design - Guide de l'utilisateur*.





Attributs :

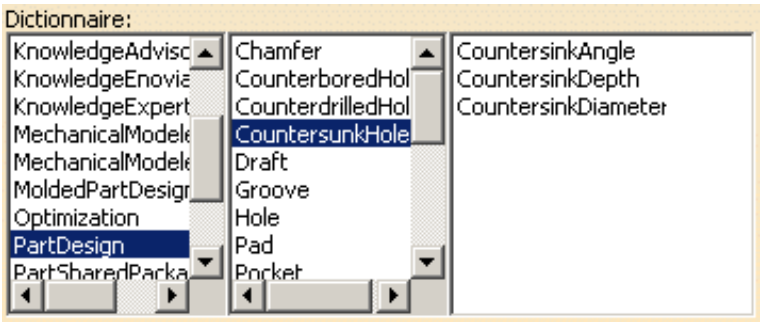
Un trou lamé chanfreiné est défini par les attributs suivants :

- CounterdrillAngle. La syntaxe à utiliser est **CounterdrillAngle = 22deg**.
- CounterdrillDiameter. La syntaxe à utiliser est **CounterdrillDiameter = 12mm**.
- CounterdrillDepth. La syntaxe à utiliser est **CounterdrillDepth** =12mm.

Trou chanfreiné

Définition :

Entité mécanique de type trou créé lorsque vous cliquez sur l'icône dans l'atelier Part Design. Pour plus d'informations, reportez-vous au document *Part Design - Guide de l'utilisateur*.





Attributs :

Un trou chanfreiné est défini par les attributs suivants :

- CountersinkAngle. La syntaxe à utiliser est **CountersinkAngle = 12deg**.
- CountersinkDepth. La syntaxe à utiliser est **CountersinkDepth** =15mm.
- CountersinkDiameter. La syntaxe à utiliser est **CountersinkDiameter** =15mm.

Cylindre

Définition :

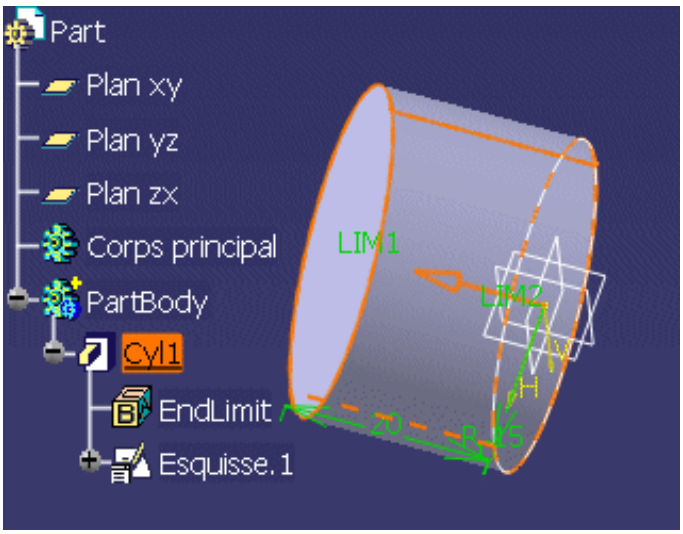
Un cylindre est une extrusion créée à partir d'une esquisse circulaire.

Attributs :

Un cylindre est défini par les attributs suivants :

- EndLimit\Length. La syntaxe à utiliser est **Length =12mm**.
- Radius : La syntaxe à utiliser est **Radius =5mm**.

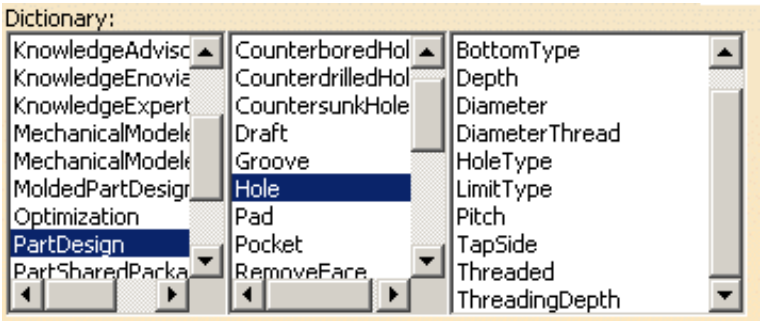
```
Cylinder1 isa CATPart
 {
  Part isa Part
   {
    PartBody isa BodyFeature
     {
      // Crée un cylindre
      Cyl1 isa Cylinder
        {
         Radius=15.0 mm;
         EndLimit=20.0 mm;
        }
     }
   }
 }
```



Trou

Définition :

Un trou est une ouverture dans une entité.



Attributs :

Un trou est défini par les attributs suivants :

- *BottomAngle* (angle de fond),
- *BottomType* (type de fond),
- *Depth* (profondeur),
- *Diameter* (diamètre),
- *DiameterThread* (diamètre de taraudage),
- *HoleType* (type de trou),
- *LimitType* (type de limite),
- *Pitch* (pas),
- *TapSide* (sens du taraudage),
- *Threaded* (taraudé),

- *ThreadingDepth* (profondeur de taraudage),
- Radius (rayon).

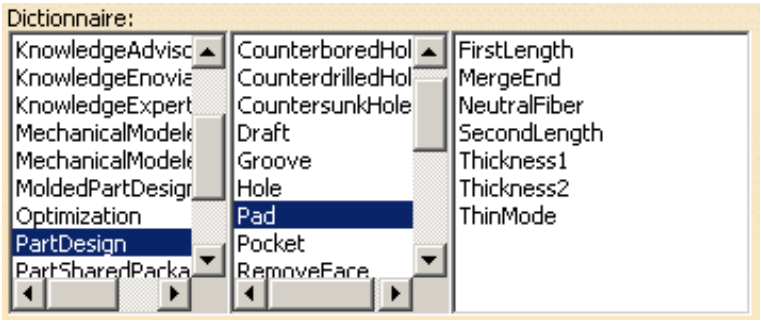
Pour spécifier un trou dans un script, vous devez utiliser l'un des trous répertoriés ci-dessous. Vous ne pouvez pas utiliser le type Hole car il correspond au type père.

- Trou lamé
- Trou chanfreiné
- Trou lamé chanfreiné
- Trou conique

Extrusion

Définition :

Une extrusion est une entité créée à partir d'une esquisse.



Attributs :

Une extrusion est définie par les attributs suivants :

- *l'esquisse* à partir de laquelle l'extrusion est réalisée,
- l'attribut *FirstLimit\Length* (ou *StartLimit\Length*)
- l'attribut *SecondLimit\Length* (ou *EndLimit\Length*).

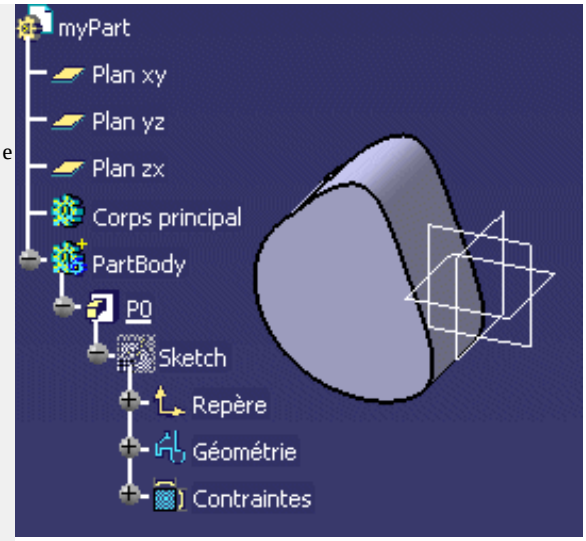
Si aucune limite n'est définie, la valeur par défaut est zéro.

```
// Utilisez la commande Chemin d'accès au fichier du
// menu contextuel pour spécifier le chemin du fichier
// à importer

import PktSketchToImport.CATPart;/*Dans le script ci-dessus,
l'extrusion P0 est créée à partir de l'esquisse Esquisse.1 importée
à partir du  document.*/

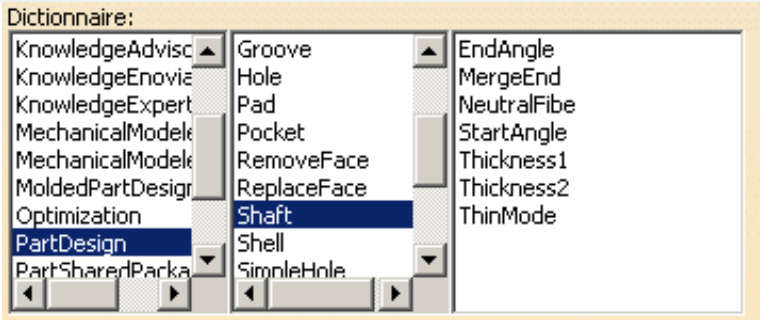


myDocument isa CATPart
 {
  myPart isa Part
   {
    PartBody isa BodyFeature
     {
      Sketch isa Esquisse.1
      {}
      P0 isa Pad("Sketch")
      {
       SecondLimit\Length=40.0mm;
      }
     }
   }
 }
```



Révolution

Une révolution est une entité créée par rotation d'une esquisse autour d'un axe.



Attributs :

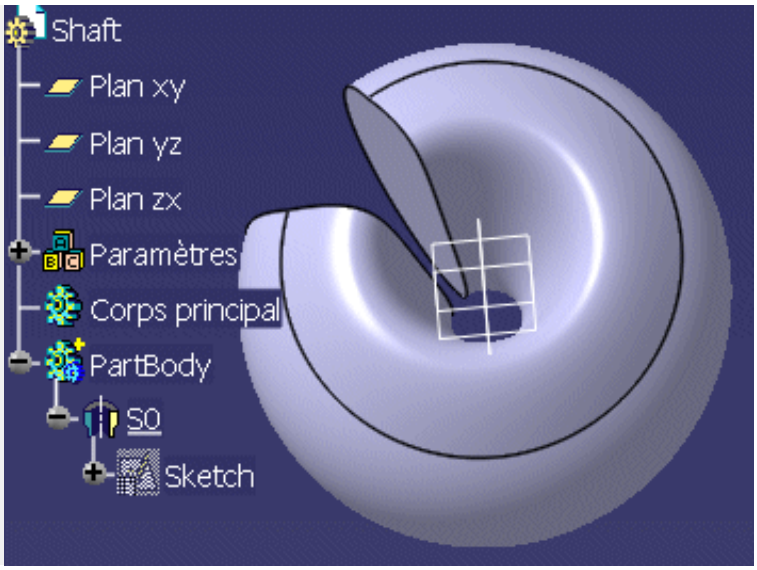
Elle se définit par deux attributs :

- *StartAngle*. La syntaxe à utiliser est **StartAngle=12deg**.
- *EndAngle* (angle de fin). La syntaxe à utiliser est**EndAngle=23deg**.
- MergeEnd
- NeutralFiber
- Thickness1
- Thickness2
- Thinmode

L'esquisse à faire pivoter doit être importée à partir d'un document CATPart externe. Ce document externe doit également contenir un axe de rotation.

```
/* Utilisez la commande Chemin d'accès au fichier du menu
contextuel pour sélectionner  l'esquisse à importer */

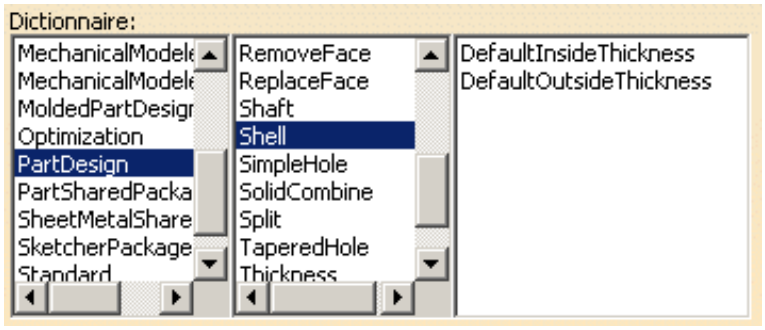

import //Utilisez la commande Chemin d'accès au fichier pour
insérer le fichier Pktsketch_shaft.CATPart .
MyShaft isa CATPart
 {
  myPart isa Part
   {
    PartBody isa BodyFeature
     {
      Sketch isa Esquisse.1 {}
      S0 isa Shaft("Sketch")
        {
         StartAngle = 20 deg ;
         EndAngle = 300 deg ;
        }
     }
   }
 }
```



Coque

Définition :

Une coque est une entité creuse.



Attributs :

Une coque est définie par les attributs suivants :

- DefaultInsideThickness. La syntaxe à utiliser est **DefaultInsideThickness = 2mm**.
- DefaultOutsideThickness : La syntaxe à utiliser est**DefaultOutsideThickness = 1mm**.

Pour définir une coque dans un script, vérifiez qu'une pièce est ouverte, puis procédez comme suit :

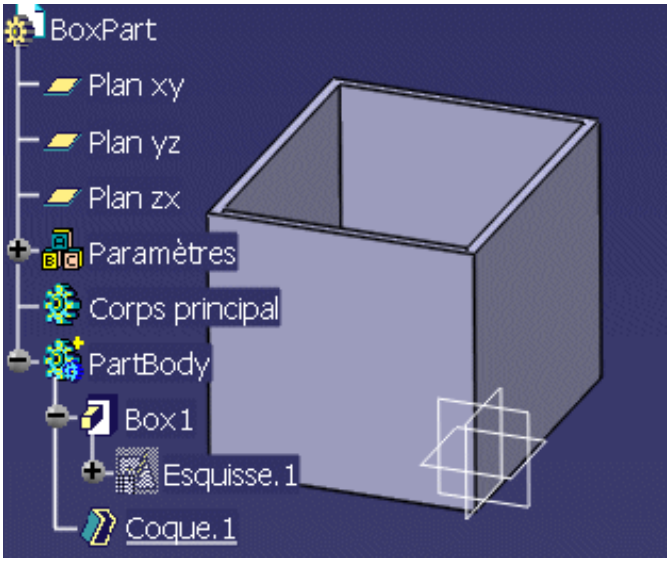
1. créez une coque à l'aide de la fonction isa

   Shell1 isa Shell ( ) { }

2. avec le bouton droit, cliquez n'importe où entre les parenthèses et sélectionnez la commande "Insérer surface" du menu contextuel. Puis, dans la zone géométrique, sélectionnez la face à creuser.

Une coque d'1mm d'épaisseur est créée par défaut.

```
MyBox isa CATPart
 {
  BoxPart isa Part
   {
    PartBody isa BodyFeature
     {
      Box1 isa Box
       {
        Width = 20.0 mm ;
        Height = 25.0 mm ;
        Length = 15.0 mm ;
       }
      Shell1 isa Shell (définition de la face)
       {
        DefaultInsideThickness = 2mm;
        DefaultOutsideThickness = 1mm;
       }
     }
   }
}
```



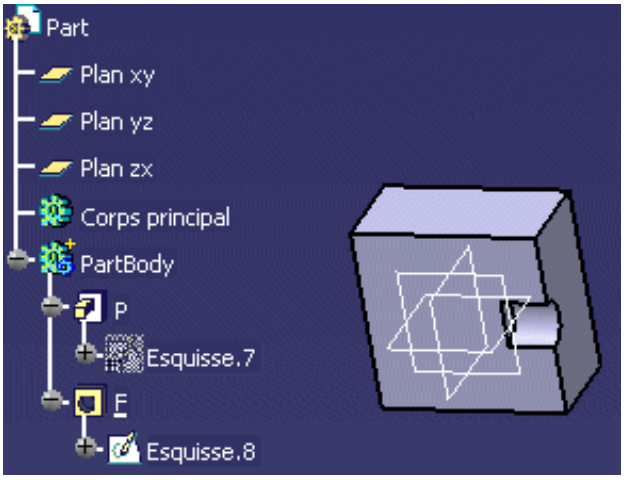
SimpleHole

Définition :

Entité mécanique de type trou créé lorsque vous cliquez sur l'icône  dans l'atelier Part Design. Pour plus d'informations, reportez-vous au document *Part Design - Guide de l'utilisateur*.

Attributs :

```
Hole1 isa CATPart
{
  Part isa Part
    {
       PartBody isa BodyFeature
       {
          P isa Pad
            {
            }
F isa SimpleHole("Utilise la commande Insérer arête du pour
sélectionner l'arête")
            {
            }
       }
    }
}
```



Sphère

Définition :

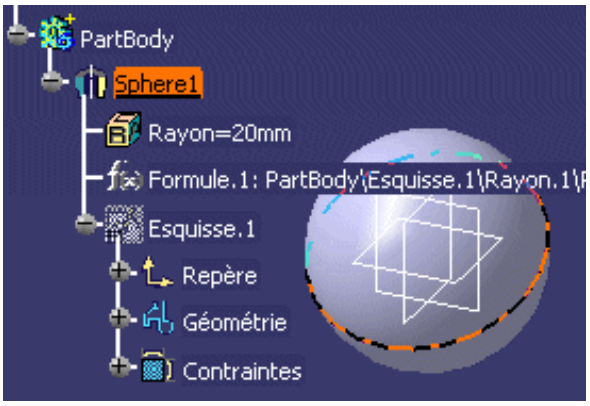
Une sphère est une révolution créée par rotation d'un demi-cercle autour d'un axe passant par les extrémités de l'arc. La seule propriété est le paramètre Radius (rayon).

Attributs :

Une sphère est définie par l'attribut suivant :

- Radius. La syntaxe à utiliser est : **Radius = 20.0 mm**.

```
MySphere  isa CATPart
 {
  SpherePart isa Part
   {
    PartBody isa BodyFeature
     {
      Sphere1 isa Sphere
        {
         Radius = 20.0 mm ;
        }
     }
   }
}
```

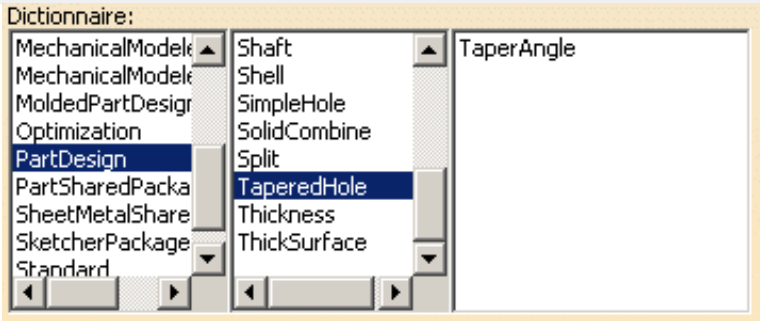


Trou conique

Définition :

Entité mécanique de type trou créé lorsque vous cliquez sur l'icône  dans l'atelier Part Design. Pour plus d'informations, reportez-vous au document *Part Design - Guide de l'utilisateur*.





Attributs

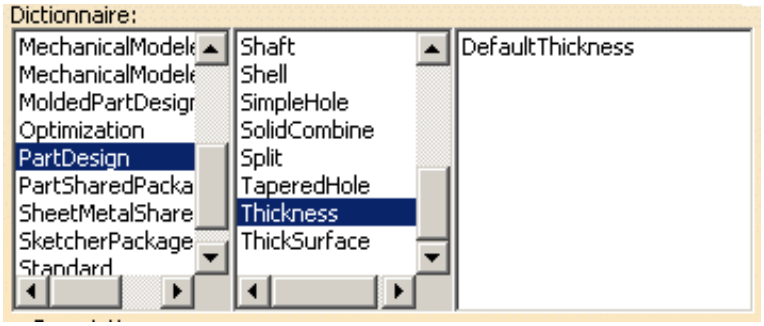
Un trou conique est défini par l'attribut suivant :

- TaperAngle:

Epaisseur

Définition :

Une épaisseur



Attributs

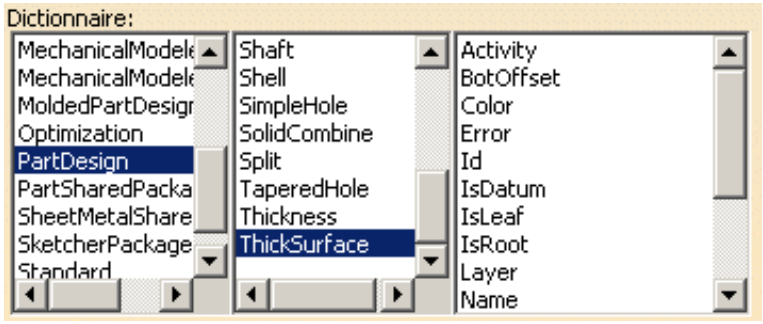
Une épaisseur est définie par l'attribut suivant :

- DefaultThickness:

Surface épaisse

Définition :

Une surface épaisse est une surface sur laquelle de la matière a été ajoutée dans deux directions opposées.



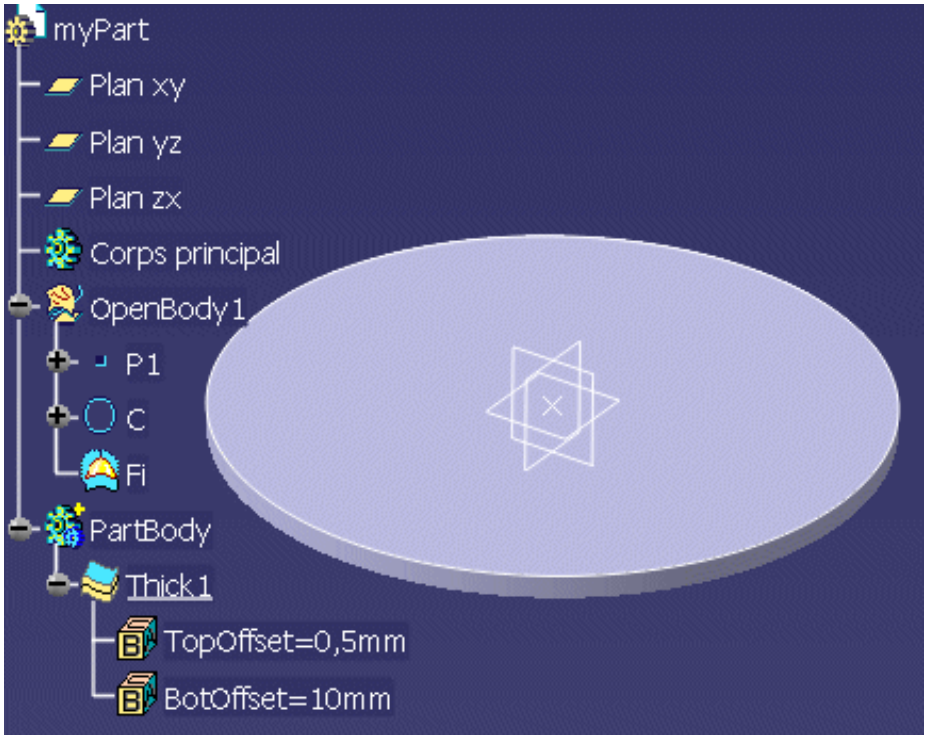
Attributs :

Une surface épaisse est définie par les attributs suivants :

- *TopOffset*, qui correspond à l'épaisseur dans une direction. La syntaxe à utiliser est**TopOffset=0.5mm**.
- *BotOffset*, qui correspond à l'épaisseur dans une direction. La syntaxe à utiliser est**BotOffset=10 mm**.

```
myThickSurface isa CATPart
 {
   myPart isa Part
     {
OpenBody1 isa OpenBodyFeature
  {
        P1 isa GSMPoint
          {
           PointType = 0;
           TypeObject isa GSMPointCoord
             {
              X = 0mm;
              Y = 0mm;
              Z = 0mm;
             }
          }
        C isa GSMCircle
          {
           CircleType = 0;
           TypeObject isa GSMCircleCtrRad
             {
               Center = object : ..\..\P1;
               Support = object : ..\..\..\`xy-
plane`;
               Radius = 150mm;
             }
            StartAngle = 0deg;
            EndAngle = 360deg;
          }
        Fi isa GSMFill
          {
           Boundary = object : ..\C;
          }
   }

       PartBody isa BodyFeature
     {
         Thick1 isa ThickSurface
   {
   TopOffset = 0.5mm;
   BotOffset = 10 mm;
   Surface = object : ..\..\OpenBody1\Fi;
       }
}
   }
}
```



Tore

Définition :

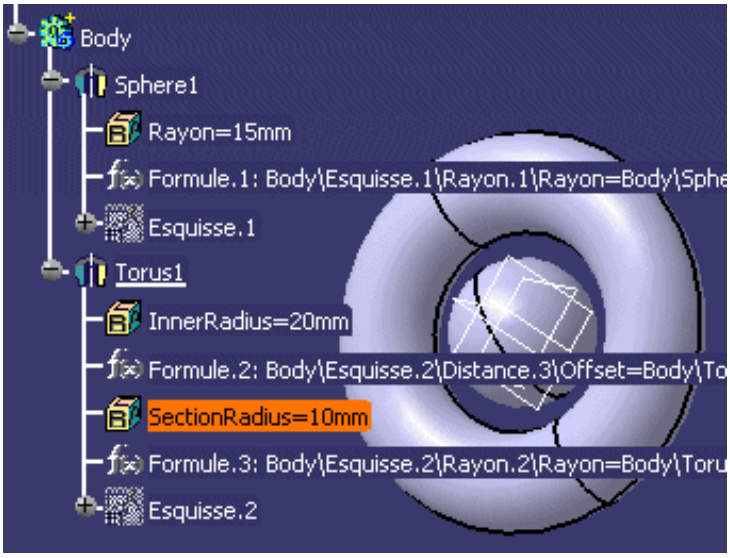
Un tore est une révolution créée par rotation d'une esquisse circulaire autour d'un axe.

Attributs :

Un tore est défini par les attributs suivants :

- *InnerRadius*
- *SectionRadius*

```
BodyDoc isa CATPart
  {
  BodyPart isa Part
    {
     Body isa BodyFeature
       {
       // Crée une sphère
       Sphere1 isa Sphere
          {
          Radius = 15.0 mm;
          }
        // Crée un tore
       Torus1 isa Torus
        {
        InnerRadius = 20.0 mm ;
        SectionRadius = 10.0 mm ;
        }
     }
    }
}
```



Module Part Shared

Fillet
ConstantEdgeFillet
Pattern

Objet Congé de raccordement constant

Définition

Un congé est une surface courbe (un arrondi) d'un rayon constant ou variable, tangent à deux surfaces et qui les touche. Ces trois surfaces réunies forment un angle externe ou un angle interne.

Remarque importante :

Pour définir un congé dans un script, vérifiez qu'une pièce est ouverte, puis procédez comme suit :

1. Créez un congé à l'aide du mot clé isa.

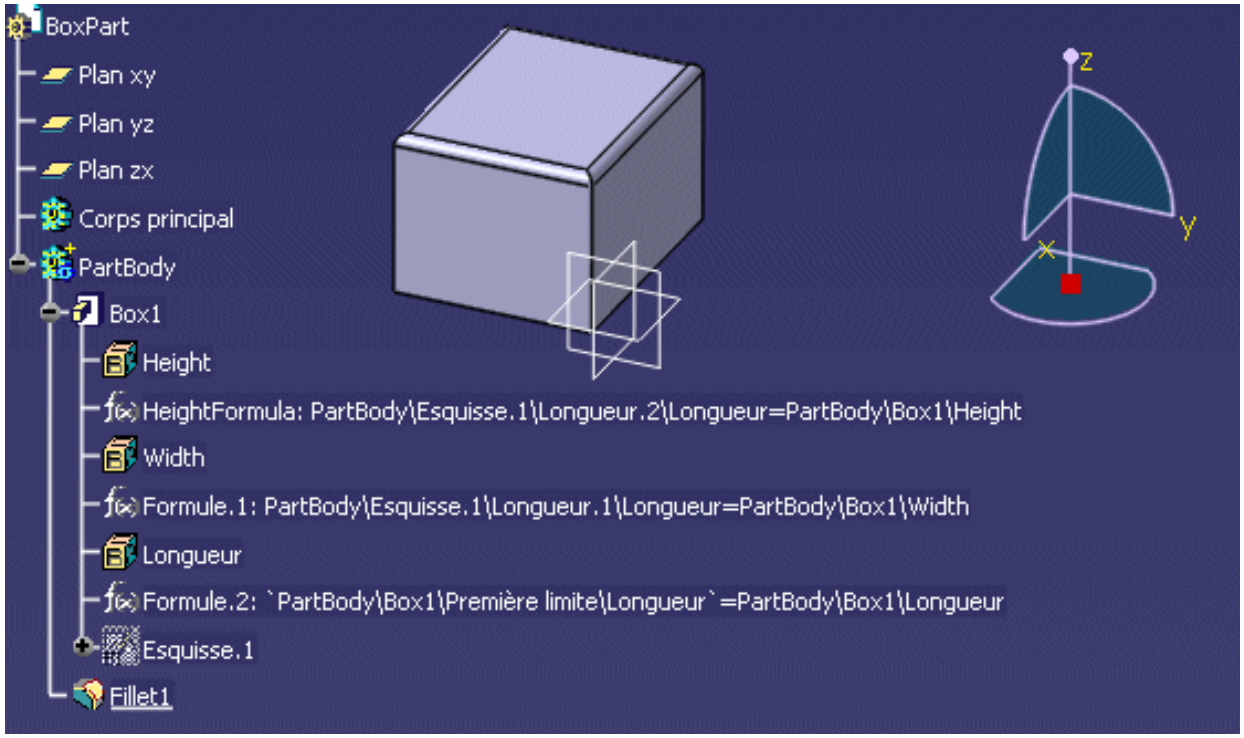
   ```
   Fillet1 isa ConstantEdgeFillet ( ) { }
   ```

2. Avec le bouton droit, cliquez n'importe où entre les parenthèses et sélectionnez la commande "Insérer arête" ou "Insérer surface" du menu contextuel. Puis, dans la zone géométrique, sélectionnez l'arête ou la face à arrondir.

Exemple

```
Box1 isa CATPart
  {
   BoxPart isa
Part
     {
      PartBody
isa BodyFeature
        {
         Box1 isa
Box
          {
           Width
= 20.0 mm ;
           Height
= 25.0 mm ;

Length = 15.0
mm ;
          }
      // Utilisez la
commande
Insérer arête ou
Insérer surface
      // du menu
contextuel pour
extraire
      // l'arête ou
la face à arrondir
          Fillet1
isa
ConstantEdgeFillet
( face à arrondir)
          {
           Radius
=1.0 mm;
          }
        }
     }
  }
```



Objet Congé

Définition

Décrit l'entité que vous créez lorsque vous cliquez sur l'icône  dans l'atelier Part Design. Pour plus d'informations, reportez-vous au document *Part Design - Guide de l'utilisateur*. Il est défini par une propriété :

- *Radius (rayon)*

Il existe trois types de congés différents :

- ConstantEdgeFillet
- FaceFillet
- TriTangentFillet

Motif

Définition

Un motif est un ensemble d'entités semblables répétées dans la même pièce. Deux types de motifs peuvent être créés avec *CATIA* : les répétitions rectangulaires et les répétitions circulaires. Actuellement, seules les répétitions rectangulaires peuvent être créées à partir d'un script. Une répétition rectangulaire se définit par les propriétés suivantes :

- Nb1, le nombre d'éléments à dupliquer le long de la première direction,
- Nb2, le nombre d'éléments à dupliquer le long de la seconde direction,
- Step1, l'espacement entre les éléments le long de la première direction,
- Step2, l'espacement entre les éléments le long de la seconde direction,
- le paramètre Activity.

Syntaxe

pattern1 isa pattern [*Nb1*,*Nb2*] of *entité_à_répéter*

Exemple

```
MyBox isa CATPart
{
BoxPart isa Part
{
PartBody isa BodyFeature
{
Box1 isa Box
{
Width = 20 mm ;
Height = 20 mm ;
Length = 20 mm ;
}

// Utilisez la commande Insérer surface du
// menu contextuel pour spécifier le point d'ancrage
// du trou
Hole1 isa SimpleHole ("Face:(Brp:(Pad.1;2);None:();Cf9:())")
{
Diameter = 15 mm;
}
Pattern1 isa Pattern[3,4] of Hole1
{
Step1 = 50 mm;
Step2 = 50 mm;
}
```

```
}
}
}
```

Module standard

Anciens noms de types	Anciens attributs	Nouveaux noms de types	Nouveaux attributs
-	-	Feature	Id Name Owner
-	-	List	-
-	-	Visualizable	Color Layer Pick Show

Module GSD Shared

Noms de types	Attributs
GSMAffinity	AxisFirstDirection AxisOrigin AxisPlane Ratio
GSMAxisToAxis	
GSMRotate	Angle Axis
GSMScaling	Ratio Reference
GSMSymetry	Reference
GSMTransformation	Activity ToTransfor
GSMTranslate	Direction Distance

Module GSD

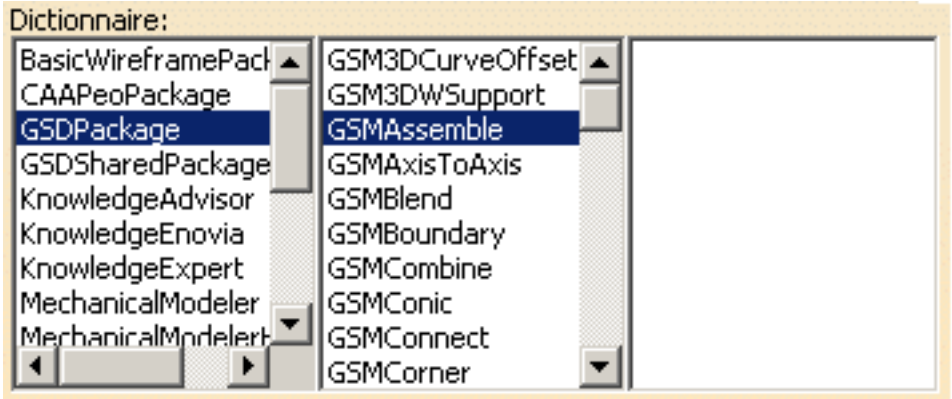
Le tableau suivant répertorie les types contenus dans le module GSD :

GSM3DCurveOffset	GSMAssemble	*GSMAxisToAxis*
GSMBlend	GSMBoundary	GSMCombine
GSMConic	GSMConnect	GSMCorner
GSMCurve	GSMCurvePar	GSMCurveSmooth
GSMCylinder	GSMDirection	GSMExtract
GSMExtractContour	GSMExtrapol	GSMExtremum
GSMExtremumPolar	GSMExtrude	GSMFill
GSMFillet	*GSMFilletBiTangent*	GSMHealing
GSMHelix	GSMIntersect	GSMInverse
GSMLawDistProj	*GSMLineCorner*	GSMLoft
GSMNear	GSMOffset	GSMProject
GSMReflectLine	GSMRevol	GSMSphere
GSMSpine	GSMSpiral	GSMSplit
GSMSweep	GSMSweepCircle	GSMSweepConic
GSMSweepSegment	*GSMSweepSketch*	GSMTrim
GSMUnfold	*GSMWSupport*	GSOBump
GSOJunction	GSOSeatDiabolo	GSPShapeMorphing
GSOVariableOffset	GSOWrapCurve	GSOWrapSurface

GSMAssemble

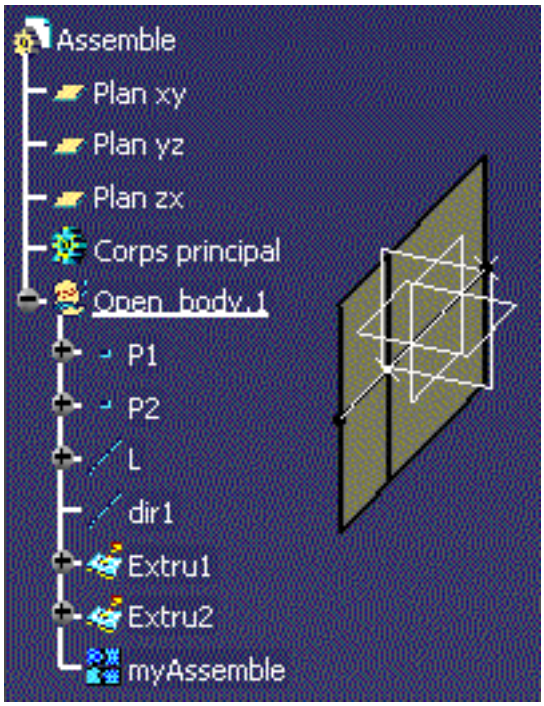
Définition :

Un objet GSMAssemble permet de joindre au moins deux surfaces ou deux courbes.
Les surfaces ou les courbes à joindre doivent être adjacentes. Pour plus d'informations, reportez-vous au document *Generative Shape Design - Guide de l'utilisateur*.



Attributs :

Cliquez ici pour ouvrir le fichier script GSMAssembleScript.

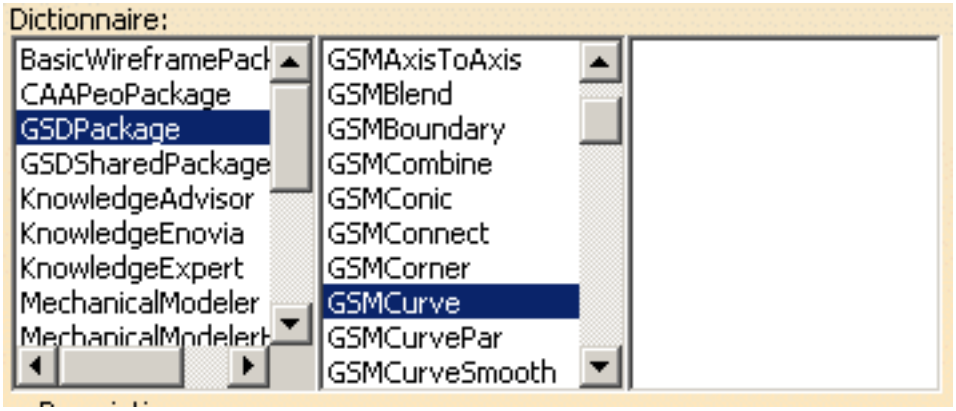


GSMCurve

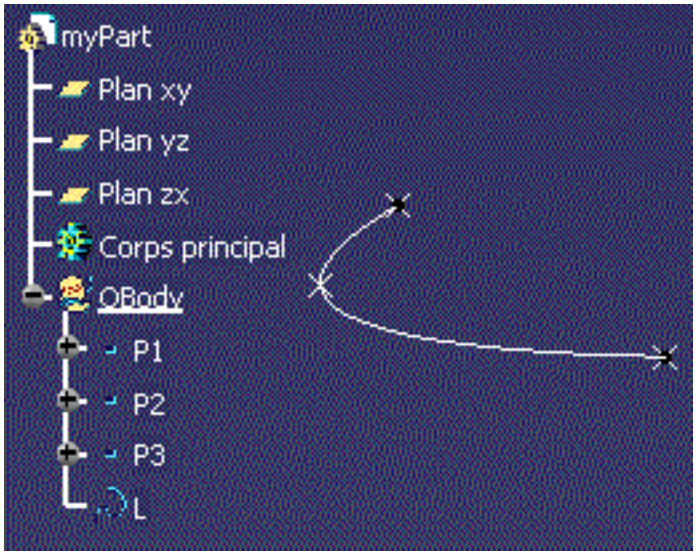
Définition :

Un objet GSMCurve est un objet créé par le programme Generative Shape Design.

Vous pouvez créer un coin en cliquant sur l'icône **Coin** () dans l'atelier Generative Shape Design.



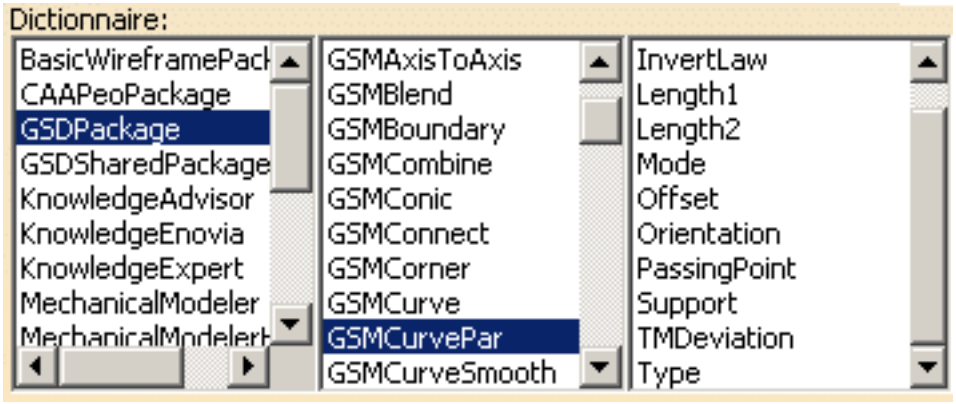
Cliquez ici pour ouvrir le fichier script GSMCurveScript.



GSMCurvePar

Définition :

Un objet GSMCurvePar est une courbe parallèle créée dans l'atelier Generative Shape Design.

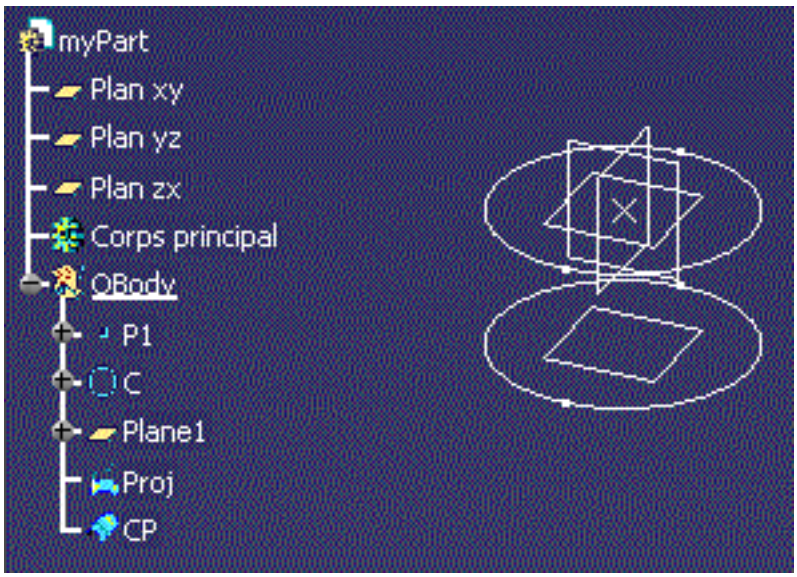


Attributs :

Un objet GSMCurvePar est défini par les attributs suivants :

- *InvertLaw*.
- *Length1*.
- *Length2*.
- *Mode*.
- *Offset*.
- *Orientation*.
- *PassingPoint*.
- *Support*.
- *TMDeviation*.
- *Type*.

Cliquez ici pour ouvrir le fichier script GSMCurveParScript.

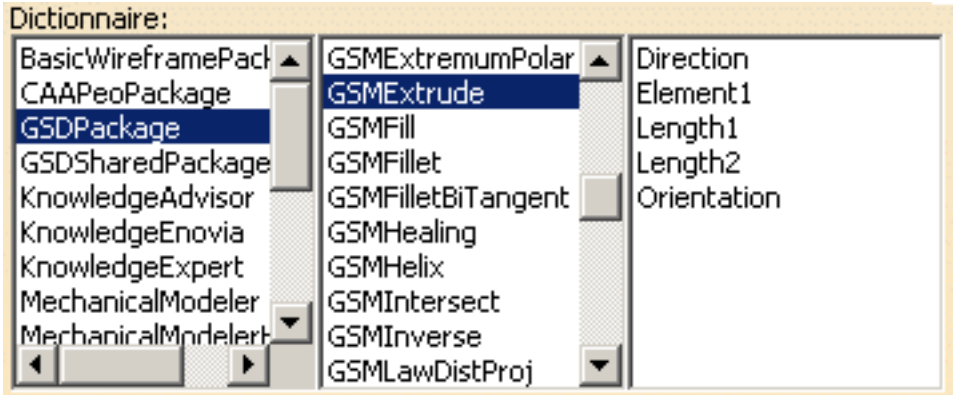


GSMExtrude

Définition :

Surface créée par l'extrusion d'un profil dans une direction donnée.

Vous pouvez créer une surface extrudée en cliquant sur l'icône **Extrusion** () dans l'atelier Generative Shape Design.

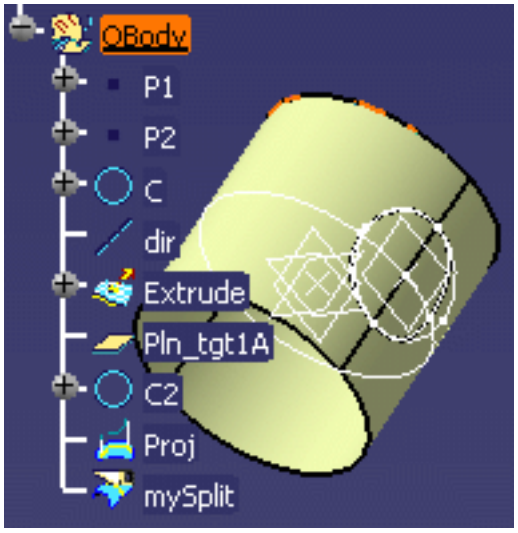


Attributs :

Un objet GSMExtrude est défini par les attributs suivants :

- *Direction*.
- *Element1*.
- *Length1*.
- *Length2*.
- *Orientation*.

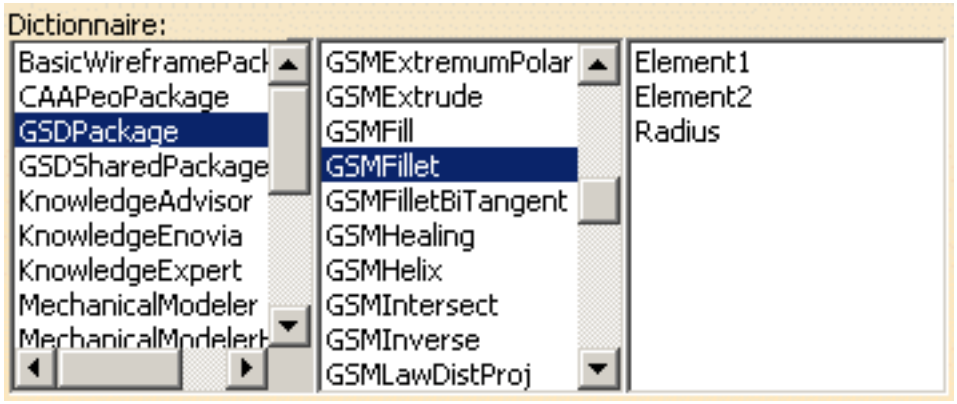
Cliquez ici pour ouvrir le fichier script GSMSplit.



GSMFillet

Définition :

Un objet GSMFillet est une surface courbe (un arrondi) d'un rayon constant ou variable, tangent à deux surfaces et qui les touche. Ces trois surfaces réunies forment un angle externe ou un angle interne.

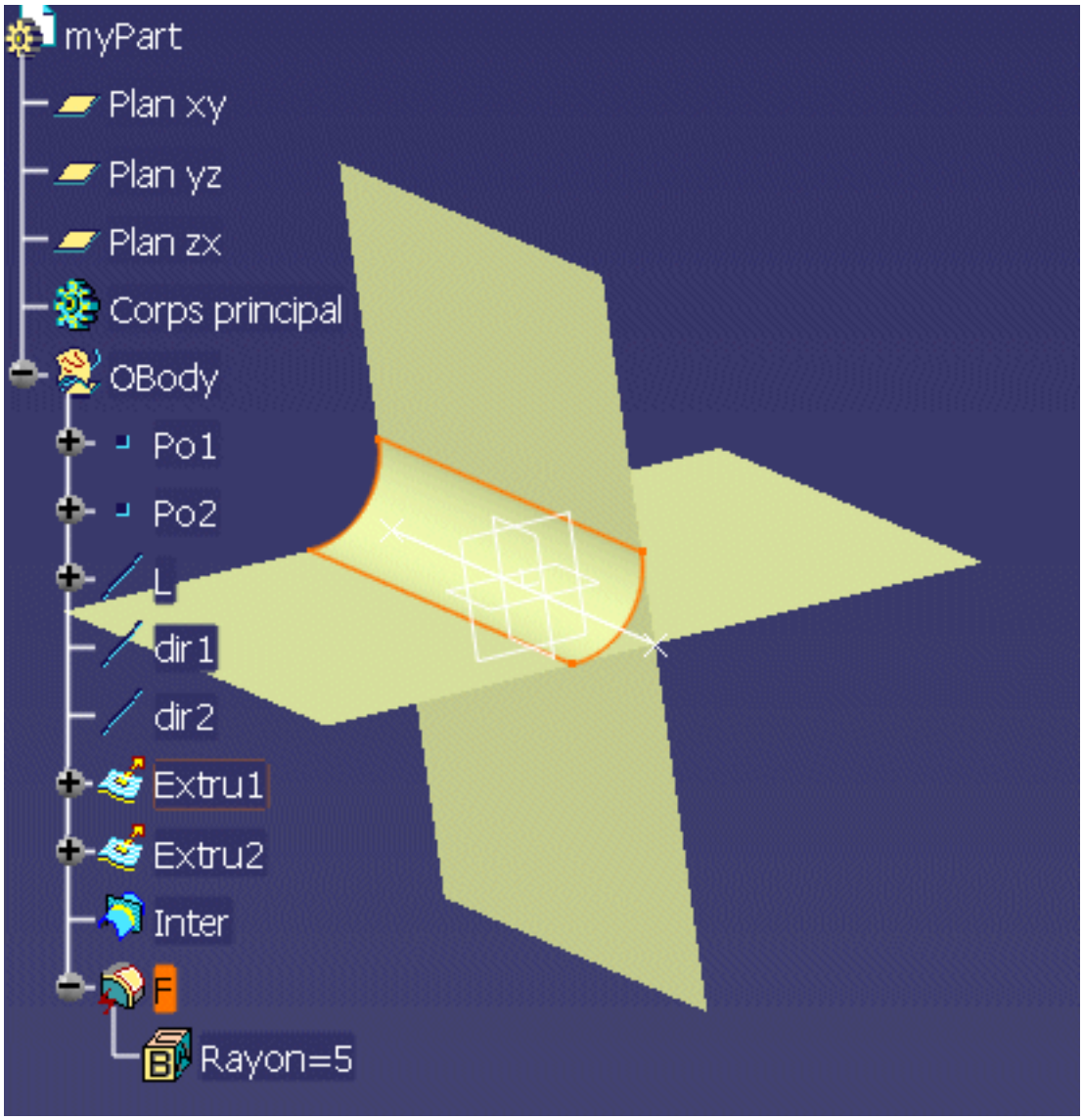


Attributs :

Un objet GSMFillet est défini par les attributs suivants :

- *Element1*.
- *Element2*.
- *Radius*.

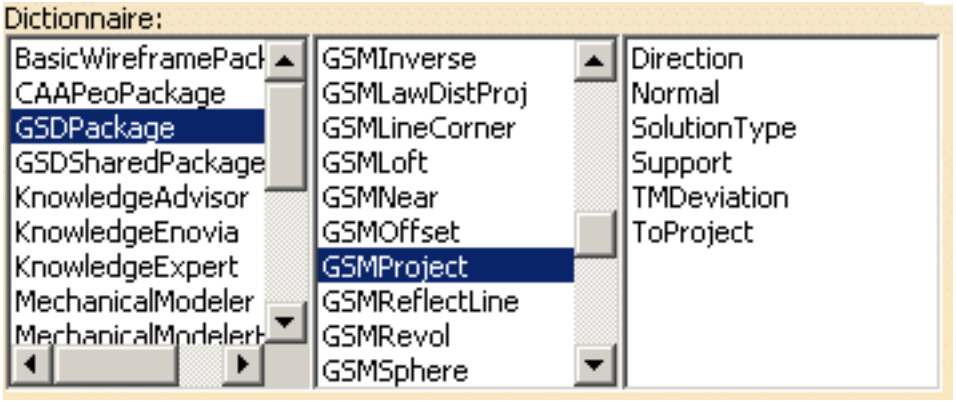
Cliquez ici pour ouvrir le fichier script GSMFilletScript.



GSMProject

Définition :

Projection Generative Shape Design. Pour plus d'informations, reportez-vous au document *Generative Shape Design - Guide de l'utilisateur*.

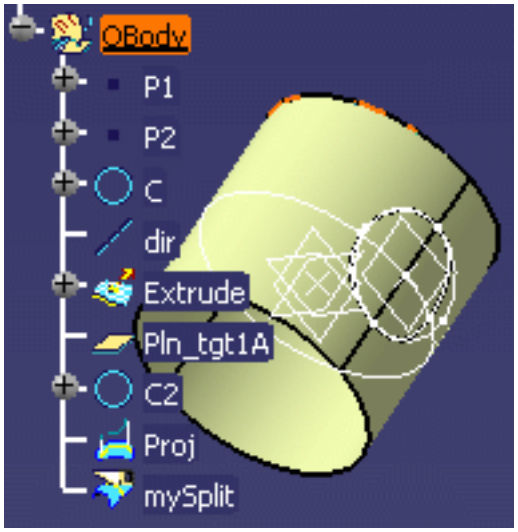


Attributs :

Un objet GSMProject est défini par les attributs suivants :

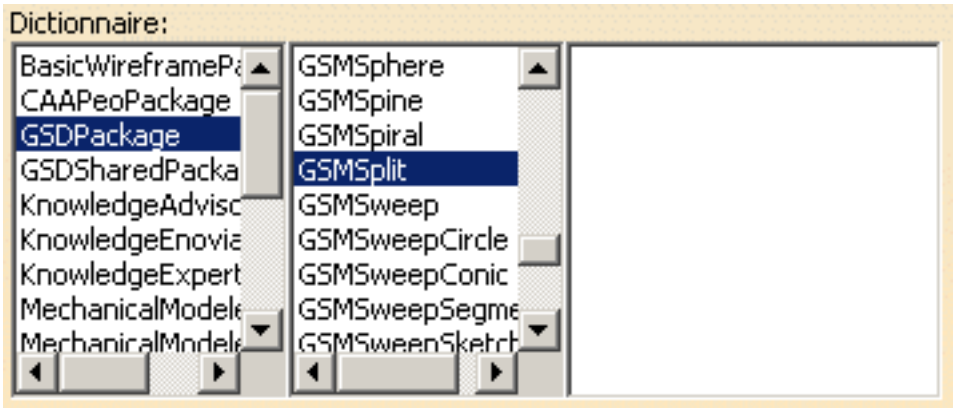
- *Direction*.
- la propriété *Normal*, qui correspond au champ du type Projection de la boîte de dialogue Définition de la projection (Normal = 1 pour une projection orthogonale ; sinon, spécifiez une direction, un objet GSMLine par exemple),
- *SolutionType*.
- *TMDeviation*.
- la propriété *ToProject*, qui correspond au champ Projeté de la boîte de dialogue Définition de la projection,
- la propriété *Support*, qui correspond au champ Support de la boîte de dialogue Définition de la projection.

Cliquez ici pour ouvrir le fichier script GSMSplit.



GSMSplit

Définition :

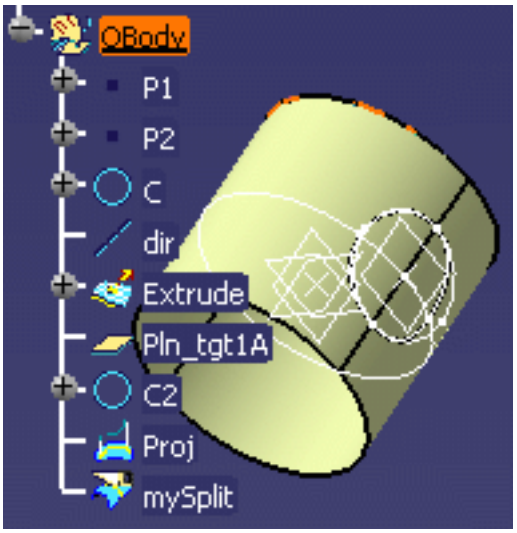


surface ou élément filaire découpé au moyen d'un élément coupant. Vous pouvez découper :

- un élément filaire par un point, un autre élément filaire ou une surface ;
- une surface par un élément filaire ou une autre surface.

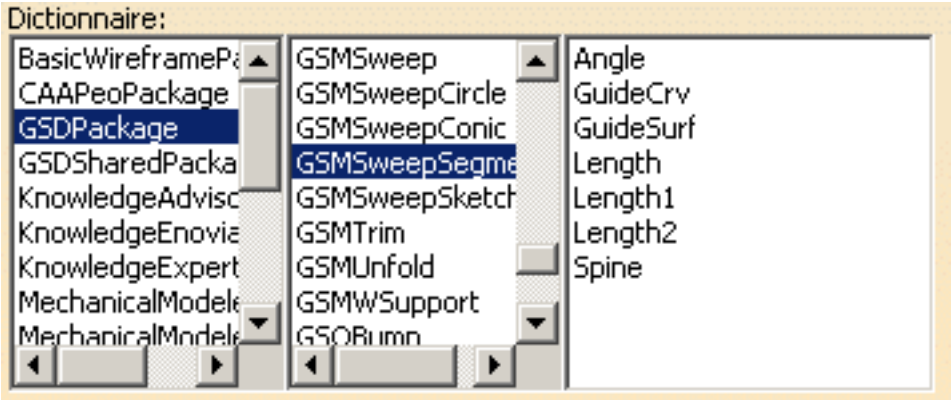
Vous pouvez découper la géométrie en cliquant sur l'icône**Coupe** () dans l'atelier Generative Shape Design.

Cliquez ici pour ouvrir le fichier script GSMSplit.



GSMSweepSegment

Définition :



Attributs :

Un objet GSMSweepSegment est défini par les attributs suivants :

- *Angle*.
- *GuideCrv*.
- *GuideSurf*.
- *Length*.
- *Length1*.
- *Length2*.
- *Spine*.

Cliquez ici pour ouvrir le fichier script GMSweepSegmentScript.

Module Knowledge Expert

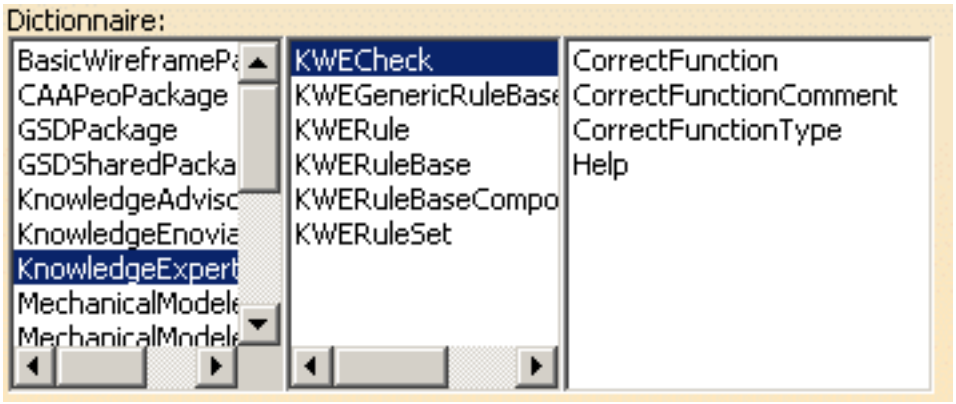
Le tableau suivant répertorie les types contenus dans le module Knowledge Expert :

KWECheck	*KWEGenericRuleBaseComponent*	KWERule
KWERuleBase	KWERuleSet	*KWERuleBaseComponent*

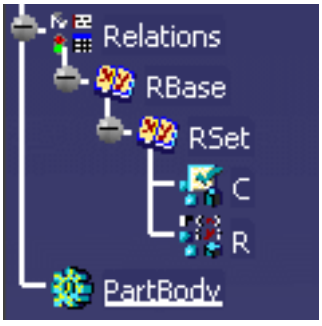
KWECheck

Définition :

Les vérifications expertes sont des fonctions générées par le produit Knowledge Expert. Elles sont regroupées en jeux de règles appartenant à une base de règles. Lorsque vous écrivez un script utilisant des vérifications, vous devez respecter la hiérarchie base de règles/jeu de règles. Pour plus d'informations sur les concepts liés aux règles et aux vérifications expertes, reportez-vous au document *Knowledge Expert - Guide de l'utilisateur*.



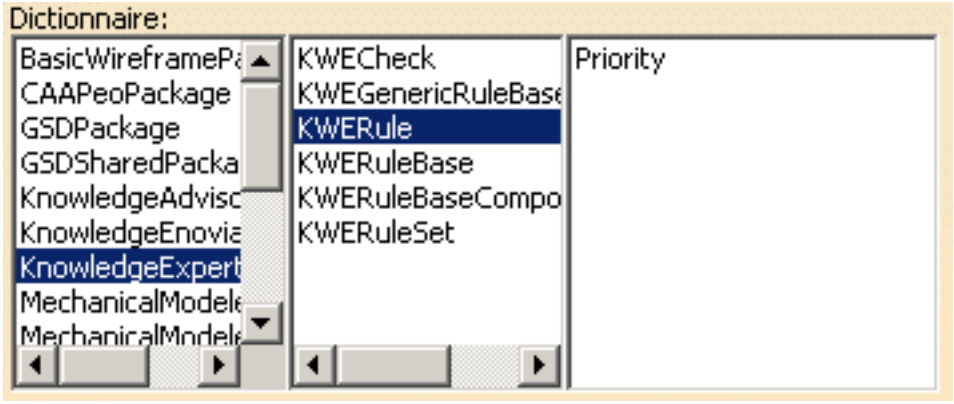
Cliquez ici pour ouvrir le fichier script KnowledgeExpertScript.



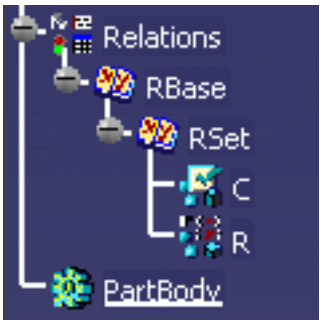
KWERule

Définition :

Les règles expertes sont des fonctions générées par le produit Knowledge Expert. Elles sont regroupées en jeux de règles appartenant à une base de règles. Lorsque vous écrivez un script utilisant des règles, vous devez respecter la hiérarchie base de règles/jeu de règles. Pour plus d'informations sur les concepts liés aux règles et aux vérifications expertes, reportez-vous au document *Knowledge Expert - Guide de l'utilisateur*.



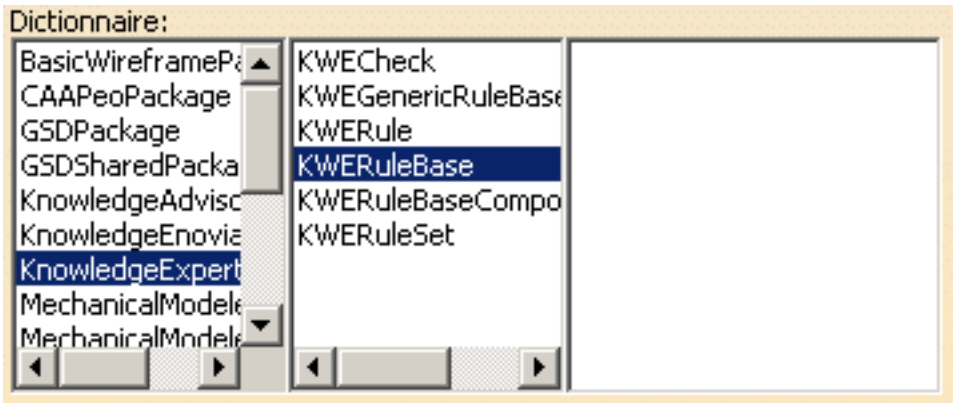
Cliquez ici pour ouvrir le fichier script KnowledgeExpertScript.



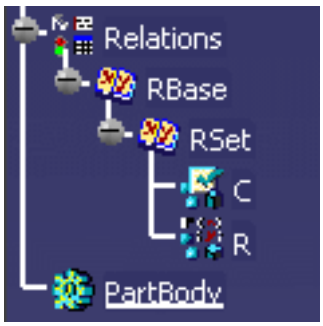
KWERuleBase

Définition :

Les bases de règles sont des entités créées avec le programme Knowledge Expert. Pour plus d'informations sur les concepts liés à ce type d'entité, reportez-vous au document *Knowledge Expert - Guide de l'utilisateur*.



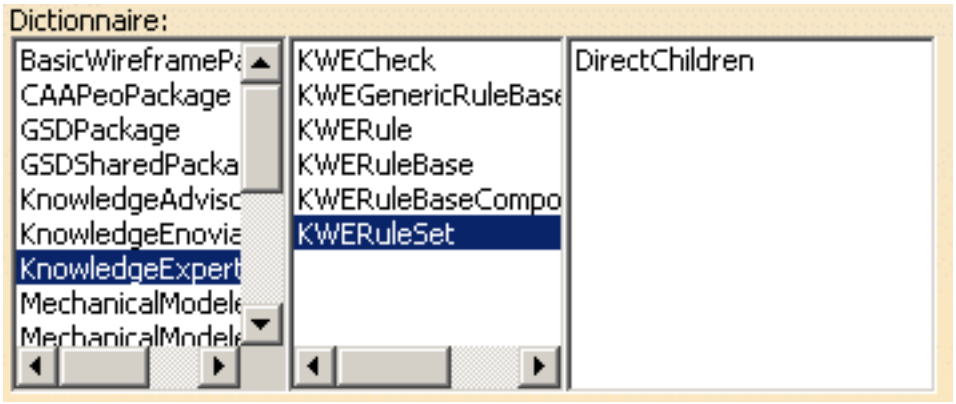
Cliquez ici pour ouvrir le fichier script KnowledgeExpertScript.



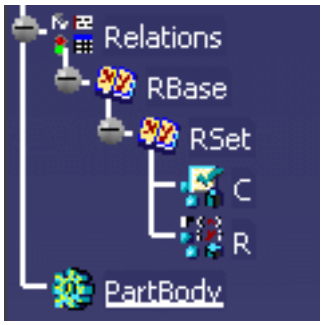
KWERuleSet

Définition :

Les jeux de règles sont des entités créées avec le programme Knowledge Expert. Pour plus d'informations sur les concepts liés à ce type d'entité, reportez-vous au document *Knowledge Expert - Guide de l'utilisateur*.



Cliquez ici pour ouvrir le fichier script KnowledgeExpertScript.



Module Modeleur mécanique

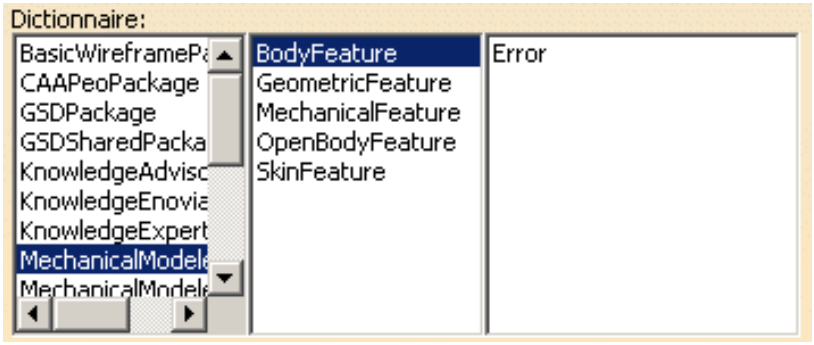
Certains types et attributs ont été modifiés. Vous trouverez ci-dessous une table de conversion répertoriant les anciens types, leurs attributs, leurs nouveaux noms (le cas échéant) et leurs attributs :

BodyFeature	*GeometryFeature*
MechanicalFeature	*OpenBodyFeature*
OpenBodyFeature	

BodyFeature

Définition :

Un corps résulte de la combinaison de plusieurs entités présentes dans une pièce. Pour plus d'informations, reportez-vous au document *Part Design - Guide de l'utilisateur*.




```
BodyDoc isa
CATPart
  {
   BodyPart isa
Part
    {
     Body isa
BodyFeature
      {
       // Crée
une sphère

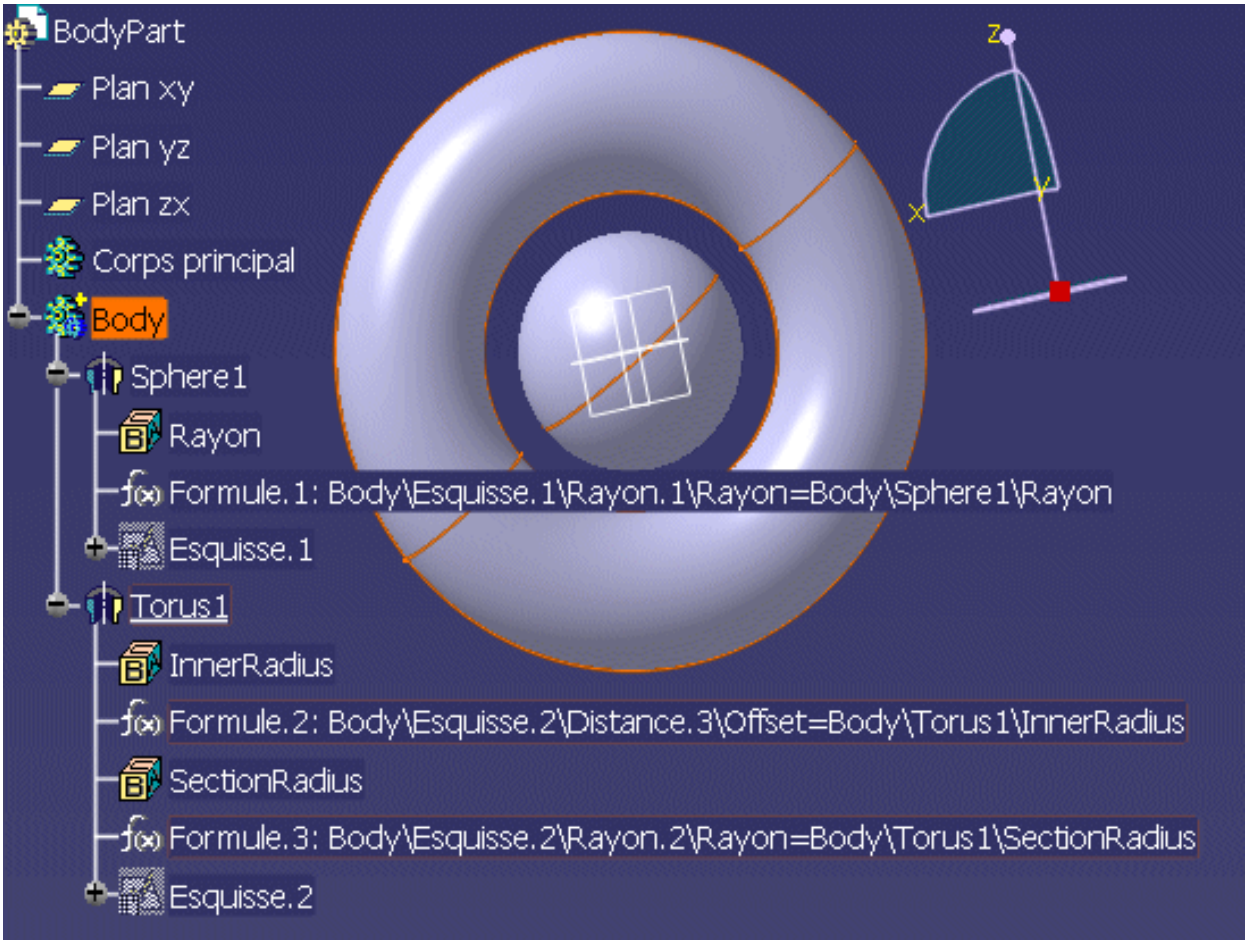
Sphere1 isa
Sphere
        {

Radius = 15.0
mm;
        }
       // Crée
un tore
       Torus1
isa Torus
        {

InnerRadius =
20.0 mm ;

SectionRadius
```

```
= 10.0 mm ;
        }
      }
    }
  }
```


BodyPart
Plan xy
Plan yz
Plan zx
Corps principal
Body
Sphere1
Rayon
Formule.1: Body\Esquisse.1\Rayon.1\Rayon=Body\Sphere1\Rayon
Esquisse.1
Torus1
InnerRadius
Formule.2: Body\Esquisse.2\Distance.3\Offset=Body\Torus1\InnerRadius
SectionRadius
Formule.3: Body\Esquisse.2\Rayon.2\Rayon=Body\Torus1\SectionRadius
Esquisse.2
Z
X
Y

Glossaire

& 9 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Symboles

.CATScript

Extension d'un fichier de macro-instructions généré par *CATIA* version 5. Vous pouvez indiquer un fichier de macro-instructions comme argument de la fonction LaunchMacroFromFile, qui peut être invoquée dans les règles et les vérifications.

.txt

Extension d'un fichier composé de caractères texte. Vous pouvez utiliser ce format de fichier d'importation pour importer des paramètres et des formules.

.xls

Extension des fichiers Excel. Vous pouvez utiliser ce format de fichier d'importation pour importer des paramètres et des formules sous Windows.

| (opérateur)

Répartit sur plusieurs lignes un message initialement contenu sur une seule. Cette option ne peut s'utiliser qu'avec la fonction Message lors de la programmation de règles et de vérifications.

A

Activité

Propriété qui définit si une relation est appliquée à un document ou non. La valeur de ce paramètre est soit vrai soit faux. Elle apparaît sous la forme d'une icône dans l'arbre de spécifications et peut être lue dans la liste des paramètres du document.

association

Liaison entre un paramètre d'un document et le paramètre équivalent dans une table de paramétrage externe. Les associations sont créées lorsque les noms des paramètres du document ne correspondent pas exactement à ceux lus dans la table de paramétrage.

C

configuration

Ligne d'une table de paramétrage. Une configuration est une série cohérente de valeurs de paramètre qui peut être appliquée à un document.

D

dictionnaire

Ensemble des paramètres, opérateurs, mots clés, fonctions et autres composants appartenant au langage utilisé pour écrire les formules, les règles et les vérifications. L'éditeur de formules, l'éditeur de règles et l'éditeur de vérifications permettent un affichage interactif du dictionnaire répertorié dans les modules pré-définis.

E

équivalence

Relation qui renseigne les entités N à l'aide d'une autre entité "pilotante". La valeur de l'entité pilotante est affectée à chaque entité de sortie.

expression

Objet (permanent ou non) utilisé pour décrire une méthode d'évaluation dans un texte et pouvant être utilisé pour réaliser l'évaluation des paramètres.

F

formule

Relation imposant une contrainte à un paramètre. La relation de la formule est une instruction composée d'une seule ligne. La première partie de la relation désigne le paramètre sujet de la contrainte tandis que la deuxième partie contient une relation utilisant d'autres paramètres comme variables. Une formule est une entité. Dans l'arbre de spécifications du document, elle apparaît sous la forme d'une relation qui peut être activée ou désactivée. Comme n'importe quelle entité, une formule peut être manipulée à partir de son menu contextuel.

R

règles de connaissance

Ensemble des programmes logiciels servant à la création et à la manipulation d'informations basées sur les connaissances. Les informations basées sur les connaissances comprennent les règles et d'autres types de relation, qui permettent aux concepteurs de garder trace des méthodes et des procédures mises en oeuvre et de les réutiliser ensuite pour d'autres travaux de conception.

T

table de paramétrage

Tableau contenant des valeurs pouvant être appliquées à un document pour gérer les valeurs de ses paramètres. Ce tableau peut être créé à partir des paramètres du document ou d'un fichier externe. Une table de paramétrage est une entité. Dans l'arbre de spécifications du document, elle apparaît sous la forme d'une relation qui peut être activée ou désactivée. Comme n'importe quelle entité, une table de paramétrage peut être manipulée à partir de son menu contextuel. La méthode d'évaluation est basée sur un tableau (fichier Excel ou .txt).

V

vérification

Série d'instructions permettant de savoir si certaines conditions sont remplies ou non. Une vérification ne modifie pas le document auquel elle s'applique. Une vérification est une entité. Dans l'arbre de spécifications du document, elle apparaît sous la forme d'une relation qui peut être activée et désactivée. Comme n'importe quelle entité, une vérification peut être manipulée à partir de son menu contextuel.

Index

&9 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

A

action

activation d'un composant

activer un paramètre externe

activité

ajout d'adresses URL

ajout d'un commentaire

ajout d'un paramètre à une arête

ajout d'un paramètre à une entité

ajout d'une ligne au fichier externe d'une table de paramétrage

application de bornes à des paramètres à l'aide d'une règle

assemblage d'objet

association d'adresses URL et de commentaires à des paramètres ou à des relations

- ajout d'adresses URL
- recherche d'une adresse URL

attributs

- Id
- Name
- NamedURLs
- Owner
- UserInfoComment

B

barre de menus

Outils

barre d'outils Outils Knowledge Advisor
barre d'outils,
Knowledge Advisor

boucle
commentaires dans le script
création d'une boucle
création d'une copie optimisée contenant une boucle
déclaration des données d'entrée
définition du contexte
fenêtre Edition
langage de script
limite
menu contextuel
modules contenus dans l'explorateur
mots clés
opérateurs
présentation
propriétés d'objet
réaction
référence
structure du script d'actions
variables

boucles
conseils utiles

C

caractère DBCS

catalogue

chemin d'accès relatif

ClashOrContact

commande

- Ajout de paramètres sur la géométrie
- Ajout d'ensemble de paramètres
- Ajout d'ensemble de relations
- Boîte à outil d'analyse des vérifications
- Chemin d'accès au fichier
- Commentaire et URL
- Déverrouille les paramètres sélectionnés
- Ensemble d'équations
- Equivalence
- Explorateur de paramètres
- Insérer arête
- Insérer axe
- Insérer élément
- Insérer surface
- Liste
- Macros avec arguments
- réactions
- Réactions
- Règle
- Table de paramétrage
- Vérification
- Verrouille les paramètres sélectionnés

commande Chemin d'accès au fichier

commande Insérer arête

commande Insérer axe

commande Insérer élément
commande Insérer surface
commentaire
commentaires
conseils utiles
boucles
formules
paramètres
réactions
règles
relations
tables de paramétrage
constantes
contrôle de la mise à jour des relations
contrôle de la synchronisation des tables de paramétrage
copie optimisée
stockage d'une table de paramétrage
copier et coller des paramètres
création de points et de droites en tant que paramètres
création d'ensembles de paramètres
création d'ensembles de relations
création d'un paramètre externe
création d'un paramètre
création d'une action Knowledge Advisor
création d'une boucle
création d'une copie optimisée contenant une boucle
création d'une formule basée sur des publications
création d'une formule
création d'une liste

création d'une réaction Knowledge Advisor
- événement AttributeModification
- événement BeforeUpdate
- événement DragAndDrop
- événement FileContentModification
- événement Insert
- événement Inserted
- événement Instantiation
- événement Remove
- événement Update
- événement ValueChange

création d'une règle
création d'une table de paramétrage à partir des valeurs des paramètres de document
création d'une table de paramétrage à partir d'un fichier existant
création d'une vérification
création et utilisation d'une loi Knowledge Advisor

D

déclaration des données d'entrée
définition du contexte
désactivation d'un composant
désactiver un paramètre externe
déverrouillage d'un paramètre
dictionnaire
- chaîne
- constantes
- direction constructeurs
- droites constructeurs
- fonction mathématique

liste
mesures de pièces
mesures
méthode evaluate
méthodes de table de paramétrage
opérateur d'analyse
opérateurs
plans constructeurs
points constructeurs
surfaces constructeurs
wireframe constructeurs

DistanceMin
DistanceMinXYZ
DistanceWireProduct

E

écriture des formules
écriture des règles et des vérifications
éditeur de vérifications
éditeur d'équations
ElecDistanceCommon
else...if
enovia lca
ensemble d'équations
entités mécaniques
équivalence
erreurs, gestion
événement

AttributeModification
BeforeUpdate
DragAndDrop
FileContentModification
Insert
Inserted
Instantiation
Remove
Update
ValueChange

événement BeforeUpdate (réaction)
événement DragAndDrop (réaction)
événement FileContentModification (réaction)
événement Insert (réaction)
événement Inserted (réaction)
événement Instantiation (réaction)
événement Remove (réaction)
événement ValueChange (réaction)
exécution d'une analyse globale des vérifications
exemple d'utilisation
roulement à billes

F

familiarisation avec la boîte de dialogue Formules
familiarisation avec la boîte de dialogue Table de paramétrage
fenêtre Edition de la liste
fonction IsSupporting
fonction LaunchMacroFromDoc

fonction LaunchMacrofromFile
fonction mathématique
fonction Message
fonction Query
fonction Question
fonction utilisateur de l'atelier Electrical
DistanceWireProduct
ElecDistanceCommon
ListAllOuterLayerObjects
ListDirectlyCoveredObjects
ListDirectlyOuterLayerObjects
VisualMode
For
formule
création d'une formule basée sur des publications
création d'une formule
familiarisation avec la boîte de dialogue Formules
présentation
référence à des paramètres externes dans une formule
spécification d'une mesure dans une formule
utilisation de la géométrie
formules
conseils utiles

G

gestion des erreurs

I

importation d'un paramètre
inside
inspecteur de gestion de connaissance
    analyse des antécédents
    analyse des dépendants
instanciation
instanciation de relations issues d'un catalogue
instructions conditionnelles
    if...else... else if
intégration avec Enovia
IsIncludedIn

L

lancement d'une macro VB avec des arguments
langage de script
langage des règles de connaissance
    commentaires
    unités
    variables temporaires
langage des règles de connaissance
let
lien associatif
lien entre des mesures et des paramètres
limite
ListAllOuterLayerObjects
ListDirectlyCoveredObjects

ListDirectlyOuterLayerObjects
Liste
loi

M

macro VB
message d'erreur de syntaxe
mesure
mesures de pièces
méthode
Extraction
Longueur
Recherche
ReplaceSubText
ToLower
ToReal
ToString
ToUpper
méthode AbsoluteId
méthode AttributeType
méthode CellAsBoolean
méthode CellAsReal
méthode CellAsString
méthode CloserInfConfig
méthode CloserSupConfig
méthode CloserValueInfInColumn
méthode CloserValueSupInColumn
Méthode d'extraction
méthode Evaluate

dictionnaire

méthode Evaluate

méthode GetAttributeBoolean

méthode GetAttributeInteger

méthode GetAttributeReal

méthode GetAttributeString

méthode HasAttribute

méthode Id

méthode IsOwnedBy

Méthode Length

méthode LocateInColumn

méthode LocateInRow

méthode MaxInColumn

méthode MinInColumn

méthode Name

méthode object

IsSupporting

méthode AbsoluteID

méthode AttributeType

méthode GetAttribute Boolean

méthode GetAttributeInteger

méthode GetAttributeReal

méthode GetAttributeString

méthode HasAttribute

méthode Id

méthode IsOwnedBy

méthode IsOwnedByString

méthode Name

méthode SetAttributeBoolean

méthode SetAttributeInteger

méthode SetAttributeReal

méthode SetAttributeString

SetAttributeDimension

méthode recherche

méthode ReplaceSubText

méthode SetAttributeBoolean

méthode SetAttributeDimension

méthode SetAttributeInteger

méthode SetAttributeReal

méthode SetAttributeString

méthode SetCell

méthode ToLower

méthode ToReal

méthode ToString

méthode ToUpper

méthodes de table de paramétrage

mise à jour de relations utilisant des mesures

mise à jour des relations

mise en route

utilisation des formules

utilisation des paramètres

utilisation des règles

utilisation des vérifications

mode Analyse des antécédents

mode Analyse des dépendants

mot clé

context

from

if...else

import

isa

publish

while

mot clé context

mot clé from

mot clé import

mot clé isa

mot clé let

mot clé publish

N

nom global

nom local

O

objet

congé constant

motif

objet balayage linéaire

objet BodyFeature

objet boîte

objet chanfrein

objet cône

objet congé constant

objet congé de raccordement

objet coque

objet courbe par
objet courbe
objet cylindre
objet de découpe
objet d'extrusion
objet épaisseur
objet extrusion
objet kwecheck
objet kwerule
objet kwerulebase
objet kweruleset
objet motif
objet projection
objet révolution
objet sphère
objet surface épaisse
objet tore
objet trou chanfreiné
objet trou conique
objet trou lamé chanfreiné
objet trou lamé
objet trou
opérateur d'analyse
opérateurs
    opérateurs arithmétiques
    opérateurs de comparaison
    opérateurs logiques

opérateurs arithmétiques
outil d'analyse
Outils
    barre de menus

Outils Options - Knowledge Advisor
onglet Paramètres et mesure
onglet Part Infrastructure

P

paramètre
activation et désactivation d'un composant
application de bornes à un paramètre à l'aide d'une règle
copie et collage de paramètres
création de points et de droites en tant que paramètres
création d'un lien entre des mesures et des paramètres
création d'un paramètre
importation d'un paramètre
nom global
nom local
publication d'un paramètre
spécification du paramètre Matériau
spécification d'une valeur de paramètre en tant que mesure
utilisation de relations basées sur les publications
verrouillage et déverrouillage d'un paramètre sélectionné

paramètre caché
paramètre constant
paramètre externe
activer
créer
désactiver
synchroniser

paramètres

conseils utiles

PenetrationMax

personnalisation des rapports de vérification

plans constructeurs

point d'interrogation dans les formules

présentation des formules

présentation des tables de paramétrage

publication d'un paramètre

publication

R

réaction

boucle

événement BeforeUpdate

événement DragAndDrop

événement FileContentModification

événement Insert

événement Inserted

événement Instantiation (gabarit)

événement Instantiation (modèle de document)

événement Remove

événement Update

événement ValueChange

utilisation de la fenêtre Réaction

utilisation de la réaction

réactions

conseils utiles

recherche d'une adresse URL
référence
Wireframe - Version de base
module Part Shared
référence à des paramètres externes dans une formule
règle
création
utilisation de la fonction Règle
utilisation de l'éditeur de règles
utilisation de règles et de vérifications dans une copie optimisée
règles
conseils utiles
relation
contrôle de la mise à jour des relations
création d'ensembles de relations
gestion des erreurs
instanciation de relations issues d'un catalogue
mise à jour de relations utilisant des mesures
utilisation de relations basées sur les publications (produit)
relation basée sur une publication
relations
conseils utiles
résolution d'un ensemble d'équations

S

sauvegarde sécurisée
spécification du paramètre Matériau
spécification d'une mesure dans une formule
stockage d'une table de paramétrage dans une copie optimisée
surfaces constructeurs
synchroniser
système de trois équations dans trois variables
résolution d'un système d'équations avec un recuit simulé
résolution d'un système d'équations avec la fonction Ensemble d'équations

T

table de paramétrage
ajout d'une ligne au fichier externe d'une table de paramétrage
conseils utiles
contrôle de la synchronisation
création d'une table de paramétrage à partir des valeurs des paramètres de document
création d'une table de paramétrage à partir d'un fichier existant
familiarisation avec la boîte de dialogue Table de paramétrage
présentation
stockage d'une table de paramétrage dans une copie optimisée
synchronisation automatique lors du chargement
synchronisation interactive lors du chargement
synchronisation manuelle

tables de paramétrage
conseils utiles

trou simple

U

unités
Update (réaction)
URL
utilisation de la fenêtre Edition de la liste
utilisation de la fenêtre Réaction
utilisation de la fonction Boucle
utilisation de la fonction Liste
utilisation de la fonction Réaction
utilisation de la fonction Règle
utilisation de la géométrie dans une formule
utilisation de la liste
utilisation de l'éditeur de règles
utilisation de l'éditeur de vérifications
utilisation de l'éditeur d'équations
utilisation de l'équivalence
utilisation de l'outil d'analyse des vérifications
utilisation de règles et de vérifications dans une copie optimisée
utilisation des relations
utilisation du dictionnaire

V

variables temporaires
    mot clé let
vérification
    création
    exécution d'une analyse globale des vérifications
    information

personnalisation des rapports de vérification
silent
utilisation de l'éditeur de vérifications
utilisation de l'outil d'analyse des vérifications
warning

vérification de type Information
vérification de type Silent
vérification de type Warning
verrouillage d'un paramètre
VisualMode

W

While
wireframe constructeurs